Logic on words

Jean-Eric Pin

LIAFA, CNRS, Université Paris VII, Case 7014
2 Place Jussieu, 75251 Paris Cedex 05, FRANCE
E-mail: Jean-Eric.Pin@liafa.jussieu.fr

Quisani: Hello, I think we met before?

Author: Yes, I visited Kevin Compton and Yuri Gurevich some time ago in Ann Arbor.

Q: I remember it. You are not a logician, aren't you?

A: No, but I would like to talk with you about questions of logic related to the theory of finite automata.

Q: I already read a column by Kevin Compton and Howard Straubing on a similar topic [13]. Are you too interested in circuits and complexity?

A: I read Howard's excellent book [42] on the subject, but it is not my main concern today. I would like to discuss more about the expressive power of fragments of Büchi's sequential calculus and related decidability questions.

Q: Would you remind me what this calculus is?

A: You identify a word $u = a_0 a_1 \ldots a_{n-1}$ on the alphabet A with a relational structure $M_u = (\mathrm{Dom}(u), <, (R_a)_{a \in A})$, where $\mathrm{Dom}(u) = \{0, \ldots, n-1\}$, $<$ is the usual order on $\mathrm{Dom}(u)$ and $R_a = \{i \in \mathrm{Dom}(u) \mid a_i = a\}$. For instance, if $u = abbaab$, then $\mathrm{Dom}(u) = \{0, 1, \ldots, 5\}$, $R_a = \{0, 3, 4\}$ and $R_b = \{1, 2, 5\}$.

Q: I thought Büchi was interested in infinite words.

A: You're right. If u is an infinite word, you simply take $\mathrm{Dom}(u) = \mathbb{N}$. You may also consider biinfinite words, for which you would take $\mathrm{Dom}(u) = \mathbb{Z}$, or words over larger ordinals than ω, but we won't consider these cases today. Now, with each sentence φ, is associated the set of words that satisfy φ:

$$\begin{aligned} L^*(\varphi) &= \{u \in A^* \mid u \text{ satisfies } \varphi\} \\ L^{\mathbb{N}}(\varphi) &= \{u \in A^{\mathbb{N}} \mid u \text{ satisfies } \varphi\} \end{aligned}$$

For instance, if $A = \{a, b\}$, the sentence

$$\varphi = \exists x\ \exists y\ \big((x < y) \wedge (R_a x) \wedge (R_b y)\big)$$

defines the languages

$$L^*(\varphi) = A^*aA^*bA^* \quad \text{and} \quad L^{\mathbb{N}}(\varphi) = A^*aA^*bA^\omega$$

It is convenient to say that two sentences φ and ψ are *$*$-equivalent* (resp. *ω-equivalent*) if $L^*(\varphi) = L^*(\psi)$ (resp. $L^{\mathbb{N}}(\varphi) = L^{\mathbb{N}}(\psi)$)

Q: I see.

A: Now an old result of Büchi [8, 9] states that a language is regular if and only if it can be defined by a monadic second order sentence. The result also holds for infinite words: just change regular into ω-regular in the statement. You remember that a set of infinite words is ω-regular if and only if it is accepted by a Büchi automaton.

Q: Or, equivalently, if it is a finite union of sets of the form XY^ω, where X and Y are regular languages. But wait a minute! Why did you consider suddenly monadic second order logic? What about full second order and first order logic?

A: You're right. Both classes are very interesting too. But monadic second order is an important border: if you consider weaker logics, such as first order, you are sure to deal with regular languages and you can hope to have easy solutions for your problems on logic by converting them into problems on finite automata. This will be the topic of our conversation today and you will see that reality is quite different, though.

If you take stronger logics, like full second order, you can define famous complexity classes like L, NL, P, NP , PH, $PSPACE$, etc. For instance, Stockmeyer [40] proved that full second order corresponds to PH, the polynomial hierarchy. There is a lot to say, and this would deserve a column by itself, but I don't want to talk about this today.

Q: All right. So you are only interested today in logics that define regular languages. I am trying to remember the proof of Büchi's theorem. Could you give me a hint?

A: Let's do it for finite words. The case of infinite words is a little more technical, because you need to consider Büchi or Muller automata instead of deterministic finite automata (dfa) but the idea is the same. The first step is to convert a regular language into a sentence by encoding the behaviour of a dfa $\mathcal{A} = (Q, A, \cdot, q_0, F)$. It suffices to associate with each state q a set variable X_q that encodes the set of positions in which a given path reaches state q. Do you see how to do it?

Q: I think so. First, let me use shortenings like Sx, the successor of x, min and max, the first and last elements of the domain, which can be easily defined by a first order formula. Next, the X_q's should be pairwise disjoint and one should have $x \in X_q$ and $Sx \in X_{q'}$ if and only if there is a letter a such that $q' = q \cdot a$. Finally, one should have $\min \in X_{q_0}$ and $\max \in \bigcup_{q \in F} X_q$. To sum up, if $Q = \{1, \dots, n\}$, the language accepted by $\mathcal{A}$ can be defined by the following sentence:

$$\exists X_1 \exists X_2 \ldots \exists X_n \left[\bigwedge_{q \neq q'} \neg \exists x \, (X_q x \wedge X_{q'} x) \wedge \forall x \bigvee_{q \cdot a = q'} \big(X_q x \wedge R_a x \wedge X_{q'} S x\big) \right.$$
$$\left. \wedge\ X_{q_0} \min \wedge \Big(\bigvee_{q \in F} X_q \max \Big) \right]$$

A: Very good. Now, we have to convert sentences into languages. The idea is to make induction on formulæ, but for this, we need first to define the language associated with a formula, and not only to a sentence. Let $x_1, \ldots, x_r$ (resp. $X_1, \ldots, X_s$) be the set of first (resp. second) order variables occurring in some formula φ. Consider a new alphabet

$$B_{p,q} = A \times \{0,1\}^p \times \{0,1\}^q$$

where $p \geq r$ and $q \geq s$. A word on $B_{p,q}$ can be identified with a sequence

$$(u_0, u_1, \ldots, u_p, u_{p+1}, \ldots, u_{p+q})$$

where $u_0 \in A^*$ and $u_1, \ldots, u_p, u_{p+1}, \ldots, u_{p+q} \in \{0,1\}^*$. Actually, we are interested in the language $K_{p,q}$ formed by all words of $u \in B^*_{p,q}$ whose components $u_1, \ldots, u_p$ contain exactly one occurrence of 1. Note that $K_{p,q}$ is a regular language. For instance, if $A = \{a, b\}$, a typical word of $B^*_{3,2}$ is represented below:

u_0	a	b	a	a	b	a	b	$\cdots$
u_1	0	1	0	0	0	0	0	$\cdots$
u_2	0	0	0	0	1	0	0	$\cdots$
u_3	1	0	0	0	0	0	0	$\cdots$
u_4	0	1	1	0	0	1	1	$\cdots$
u_5	1	1	0	1	0	1	0	$\cdots$

Figure 1: A word of $B^{\omega}_{3,2}$.

Now the predicate R_a is interpreted as the set of positions carrying an a in u_0, each set variable X_j as the set of positions carrying a 1 in u_{p+j}, and the first order variables x_j as the unique position carrying a 1 in u_j.

Q: So if $p = q = 0$, you have $B_{p,q} = A$ and you obtain the interpretation of sentences you had before.

A: Right.

Q: Let me try for myself to interpret the formulæ. If φ is atomic, this is easy. For instance

$$L^*(x_i < x_j) = K_{p,q} \cap B^*_{p,q} C_i B^*_{p,q} C_j B^*_{p,q}$$

where $C_k = \{b \in B_{p,q} \mid b_k = 1\}$ and

$$L^*(X_i x_j) = K_{p,q} \cap B_{p,q}^* C_{i+p,j} B_{p,q}^*$$

where $C_{i,j} = \{b \in B_{p,q} \mid b_i = b_j = 1\}$. Disjunction corresponds to union and negation to complement.

A: Not exactly. Remember that the universe is now $K_{p,q}$.

Q: All right. So $L^*(\neg\varphi) = K_{p,q} \setminus L^*(\varphi)$: negation corresponds to the operation $L \to K_{p,q} \setminus L$. Since $K_{p,q}$ is regular, this operation still preserves regular languages. I think I can finish the proof now. Formulae of the form $\exists x \varphi$ and $\exists X \varphi$ correspond to projections.

A: Exactly. If you denote by $\pi_i : B_{p,q} \to B_{p-1,q}$ the projection that erases the i-th component, defined by

$$\pi_i(b_0, b_1, \ldots, b_{p+q}) = (b_0, b_1, \ldots, b_{i-1}, b_{i+1}, \ldots, b_{p+q})$$

then $L^*(\exists x_i \varphi) = \pi_i(L^*(\varphi))$ and $L^*(\exists X_i \varphi) = \pi_{p+i}(L^*(\varphi))$. Now it is well-known that regular languages are closed under morphism.

Q: Let me come back to my first question. What happens if you just consider first order logic?

A: The question was answered by McNaughton and Papert [18] for finite words and by Thomas [45] for infinite words. What you get is the class of *star-free* languages. Let us consider the case of finite words first. One can define the star-free languages as the smallest class of languages containing $\emptyset$, $\{1\}$, where 1 is the empty word, the languages $\{a\}$ for $a \in A$ and closed under finite union, complement and (concatenation) product. I have to warn you that this definition is a little tricky. Take $A = \{a, b\}$ and try to see which of the following languages are star-free:

(1) A^*	(2) a^*	(3) $(ab)^*$	(4) $(aa)^*$
(5) $(a(ab)^*b)^*$	(6) $(a, bab)^*$	(7) $(ab, ba)^*$	

Q: Let's see. The language A^* is the complement of the empty set, thus it is star-free. This seems to be a good trick: a^* is star-free if and only if its complement is star-free ... But this complement is A^*bA^*, the set of words containing at least one occurrence of b and it is star-free, as the product of three star-free languages, A^*, b and A^*.

A: You're pretty good!

Q: Let's try with $(ab)^*$. Its complement is $bA^* \cup aA^* \cup A^*aaA^* \cup A^*bbA^*$ and so it is star-free. The language $(aa)^*$ looks suspiciously similar, so there must be a trap ...

A: You're right again. The language $(aa)^*$ is *not* star-free. I give you the reason in a minute. Do you want to try the remaining examples?

Q: I give up; they look quite difficult and I don't see any general algorithm. Is there any?

A: There is one, discovered by Schützenberger in 1965 [36]. But I need a little bit of algebra to explain this result. Do you know anything about semigroups and monoids?

Q: I think I remember the definition. A semigroup is a set equipped with an associative multiplication and a monoid is a semigroup with identity. I also know what a semigroup morphism is. Given two semigroups S and T, a *semigroup morphism* from S into T is a map $\varphi : S \to T$ such that, for all $x, y \in S$, $\varphi(xy) = \varphi(x)\varphi(y)$. For monoid morphisms, the condition $\varphi(1) = 1$ is also required. I am afraid that's all I know about the subject.

A: Fair enough! It is sufficient to give the main definition. A monoid M *recognizes* a language L of A^* if there exists a surjective monoid morphism $\varphi : A^* \to M$ and a subset P of M such that $L = \varphi^{-1}(P)$. A language is *recognizable* if it is recognized by a finite monoid. One can show that a language is regular if and only if it is recognizable.

Q: So finite monoids are in some sense equivalent to finite dfa. Is it possible to pass from one world to the other?

A: It's fairly easy. If you have a finite dfa $\mathcal{A}$, each letter defines a function from the set of states into itself. Now take the monoid generated by these functions under the composition of functions. This finite monoid, called the *transition monoid of the automaton*, recognizes the language accepted by $\mathcal{A}$. Conversely, if there exists a surjective monoid morphism $\varphi : A^* \to M$ and a subset P of M such that $L = \varphi^{-1}(P)$, the automaton $(M, A, \cdot, 1, P)$, where $m \cdot a = m\varphi(a)$ for all $a \in A$ and $m \in M$, accepts L.

Q: Is there something similar to the notion of minimal automaton?

A: Yes, the *syntactic monoid*, which is the transition monoid of the minimal automaton. But there is a better definition. Given two monoids M and N, let us write $M \leq N$ if there is a surjective morphism from N onto M. Then $\leq$ is a quasiorder on monoids (and even an order on isomorphic types of finite monoids). Now the syntactic monoid of a recognizable language L is the smallest monoid (for $\leq$) among the monoids recognizing L. Finally, there is a third, direct, definition. Define a relation $\leq_L$ on A^* by setting $x \leq_L y$ if and only if, for every $u, v \in A^*$, $uyv \in L$ implies $uxv \in L$. Now set $x \sim_L y$ if $x \leq_L y$ and $y \leq_L x$. This defines a congruence on A^* called the *syntactic congruence* and the syntactic monoid is the quotient of A^* by this congruence.

Q: I understand these definitions, but what is the use for them?

A: Star-free sets are characterized by a simple algebraic property of their syntactic monoid M: there exists an integer n such that, for every $x \in M$, $x^n = x^{n+1}$. Monoids satisfying this property are called *aperiodic*.

Q: Wow! Let me try with $(aa)^*$. You told me it was not star-free so its syntactic monoid should not be aperiodic. Let me draw its minimal automaton:

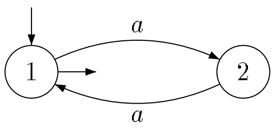


Figure 2: The minimal automaton of $(aa)^*$.

If I follow your algorithm, its syntactic monoid M is generated by the partial functions a and b. But b is the function 0 with empty domain, and a is the transposition (1 2): its square is the identity function 1. Thus M has only three elements, 1, a and 0, and $a^2 = 1$. In particular, there are no relations of the form $a^n = a^{n+1}$ for any n, so M is not aperiodic. What happens with $(ab)^*$? Here is the minimal automaton ...

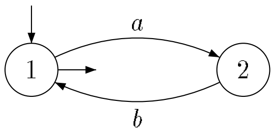


Figure 3: The minimal automaton of $(ab)^*$.

Since $(ab)^*$ is star-free, its syntactic monoid should be aperiodic, shouldn't it? I obtain five elements: 1, a, b, $aa = 0$, ab and ba, which satisfy the relations $a^2 = b^2 = 0$, $aba = a$ and $bab = b$. Now $a^2 = a^3$, $b^2 = b^3$, $(ab)^2 = ab$ and $(ba)^2 = ba$, all right, it is aperiodic. But even if you know that the syntactic monoid is aperiodic, how do you find a star-free expression?

A: The algorithm is hidden in the proof. You should read Perrin's chapter on automata in [24] to have the details.

Q: Still, it looks difficult to compute the syntactic monoid of a language. Is there any better algorithm for determining whether a language is star-free?

A: You don't need to compute the syntactic monoid. It suffices to check whether the minimal automaton of the language is aperiodic, or "counter-free". A dfa has a *counter* if there exists a sequence of pairwise distinct states $q_0, \ldots, q_n$ (with $n > 0$) and paths *with the same label* from q_0 to q_1, q_1 to q_2, ..., q_{n-1} to q_n and q_n to q_0. On the other hand, Cho and Huynh proved that finite automaton aperiodicity is PSPACE-complete [10].

Q: I think I can now check by myself whether a language is star-free. Let's go back to logic.

A: Schützenberger and McNaughton theorems lead to the following characterization: a language can be defined by a first order sentence if and only if its syntactic monoid is finite and aperiodic.

Q: And thus one can decide whether a monadic second order sentence is $*$-equivalent to a first order sentence.

A: Exactly.

Q: Could you explain me why star-free languages can be expressed in first order logic?

A: The proof is by induction on the formation rules. The key argument concerns the product: if X is defined by φ and Y is defined by ψ, then XY can be defined by $\exists x\,(\varphi'(\min, x) \wedge \psi'(Sx, \max))$, where $\varphi'(u, v)$ is the relativization of φ to the elements t such that $u \leq t \leq v$.

Q: And conversely, how do you prove that the language defined by a first order sentence is star-free?

A: This is more difficult. You would enjoy the outline given by Thomas in [48]: Fraïssé-Ehrenfeucht games are one of the main ingredients of this proof. But you can also try to mimic the proof of Büchi's theorem we did before. I will help you.

Q: Let's see. I adopt your notations, but since I have no second order variable, I can get rid of the index q. So I can interpret all first order formulæ with at most p free variables on B_p^*, where $B_p = A \times \{0,1\}^p$. The universe is K_p, the set of all words u of B_p^* whose components $u_1, \ldots, u_p$ contain exactly one occurrence of 1.

A: Do you see why K_p is star-free?

Q: I think so. If C_i denotes the set of $b \in B_p$ such that $b_i = 1$, one has

$$K_p = \bigcap_{1 \leq i \leq p} (B_p^* C_i B_p^* \setminus B_p^* C_i B_p^* C_i B_p^*)$$

The languages associated with atomic formulæ are also star-free. Disjunction and negation are easy too. For the existential quantifiers, there is this formula $L^*(\exists x_i \varphi) = \pi_i(L^*(\varphi))$. It would be nice if star-free languages were closed under morphisms.

A: Unfortunately, they are closed under inverse morphisms but not under morphisms. Some more work is needed to achieve the proof. Set $L = L^*(\varphi)$. Note that every word $x \in K_p$ has a unique decomposition of the form $x = x'bx''$, where $b \in C_i$.

Q: I see. You just locate the unique occurrence of 1 in x_i.

A: Now, for every $x \in A^*$, set

$$L(x) = \{y \in A^* \mid \forall z \in A^*\ (yz \in L \iff xz \in L)\}$$
$$R(x) = \{y \in A^* \mid \forall z \in A^*\ (zy \in L \iff zx \in L)\}$$

I claim that

$$L = \bigcup L(x')bR(x'') \qquad (*)$$

where the union runs over the set E of all triples (x', b, x'') such that $x', x'' \in B_p^*$, $b \in C_i$ and $x'bx'' \in L$.

Q: Let's see. Let L' be the right hand side of $(*)$. The inclusion $L \subseteq L'$ is obvious since $x' \in L(x')$ and $x'' \in R(x'')$. For the opposite inclusion, let $(x', b, x'') \in E$, $u' \in L(x')$ and $u'' \in R(x'')$. Then $x'bx'' \in L$ by definition of E, $u'bx'' \in L$ since $u' \in L(x')$ and finally $u'bu'' \in L$ since $u'' \in R(x'')$. Thus $L' \subseteq L$.

A: Well done.

Q: Formula $(*)$ gives a decomposition of L as union of products, but this union is infinite!

A: This is the main trick. I let you verify as an exercise that

$$L(x) = (\bigcap_{y \in x^{-1}L} Ly^{-1}) \setminus (\bigcup_{y \notin x^{-1}L} Ly^{-1})$$

where $Ly^{-1} = \{u \mid uy \in L\}$. A dual formula holds for $R(x)$. You know that, for a regular language L, there are only finitely many quotients of the form Ly^{-1} (resp. $y^{-1}L$), don't you?

Q: Yes, it's a well known result.

A: It follows that there are finitely many sets of the form $L(x)$ and $R(x)$ and so, the apparently infinite union of $(*)$ reduces to a finite union. It is also easy to see that star-free sets are closed under quotients and thus the languages $L(x)$ and $R(x)$ are star-free.

Q: I understand. So now

$$L^*(\exists x_i \varphi) = \pi_i(L^*(\varphi)) = \bigcup \pi_i(L(x')\, b\, R(x'')) = \bigcup \pi_i(L(x'))\, \pi_i(b)\, \pi_i(R(x''))$$

But all the letters of the i-th component of a word in $L(x')$ (resp. $R(x'')$) are 0's. So if $\delta_i : B_{p-1} \to B_p$ is the morphism defined by

$$\delta_i(b_0, \ldots, b_{p-1}) = (b_0, \ldots, b_{i-1}, 0, b_{i+1}, \ldots, b_p)$$

one has $\pi_i(L(x')) = \delta_i^{-1}(L(x'))$ and $\pi_i(R(x'')) = \delta_i^{-1}(R(x''))$. It follows, since star-free languages are closed under inverse morphisms, that $\pi_i(L(x'))$ and $\pi_i(R(x''))$ are star-free. Therefore $L^*(\exists x_i \varphi)$ is star-free.

A: That's it.

Q: What happens for infinite words? You mentioned that Thomas extended the theorem of McNaughton-Papert, didn't you?

A: Yes I did. You first need to define star-free ω-languages as the finite unions of sets of the form XY^ω where X and Y are star-free and $Y^+ = Y$. Equivalent definitions are also possible [17]. Thomas established that a set of infinite words is first-order definable if and only if it is a star-free ω-language [45, 46].

Q: Is there any notion of syntactic monoid in this case?

A: Arnold [2] defined a syntactic congruence for infinite words. But it took some time to find the right algebraic framework [22, 23, 19, 20, 49, 26]. First, it is more appropriate to work with finite and infinite words at the same time. Next, monoids should be replaced by ω-semigroups.

Q: What is this?

A: An ω-semigroup is a two-sorted algebra $S = (S_f, S_\omega)$ equipped with three operations: a product $S_f \times S_f \to S_f$, an infinite product $S_f^{\mathbb{N}} \to S_\omega$ and a mixed product $S_f \times S_\omega \to S_\omega$. All these products are associative and compatible in any natural sense. In particular, an infinite product $s_0 s_1 s_2 \cdots$ is unchanged if you replace a finite subsequence $s_i s_{i+1} \cdots s_j$ by its product. Note that, for any alphabet A, the pair $A^\infty = (A^+, A^{\mathbb{N}})$ is an ω-semigroup under concatenation. Morphisms are defined in a natural way. Now an ω-semigroup $S = (S_f, S_\omega)$ recognizes a subset $X = (X_f, X_\omega)$ of $(A^+, A^{\mathbb{N}})$ if there exist a morphism $\varphi : S \to A^\infty$ and a subset $P = (P_f, P_\omega)$ of S such that $X = \varphi^{-1}(P)$ (that is, $X_f = \varphi^{-1}(P_f)$ and $X_\omega = \varphi^{-1}(P_\omega)$.) A subset X of A^∞ is *recognizable* if it is recognized by a finite ω-semigroup [27, 26, 32, 49].

Q: What do you mean by a "finite" ω-semigroup?

A: An ω-semigroup $S = (S_f, S_\omega)$ such that S_f and S_ω are finite sets.

Q: Hold on ! It is not really a finite object, isn't it? You need an infinite table to define your infinite product.

A: There is a trick to overcome this problem. A Ramsey-type argument shows that finite ω-semigroups are totally determined by the finitary product, the mixed product and the ω-power (infinite products of the type $s^\omega = ssss\cdots$).

Q: I see another problem. There is no general notion of minimal automaton for infinite words. How can you define a syntactic ω-semigroup?

A: You're right, the first definition of a syntactic semigroup can't be generalized to the case of infinite words. But the two other ones can be adapted. In particular, given any finite Büchi automaton accepting a language L, you can effectively compute an ω-semigroup recognizing L and then compute the syntactic ω-semigroup.

Q: Has anybody extended the theorem of Schützenberger to infinite words?

A: Perrin did. A set of infinite words is star-free if and only if its ω-semigroup is aperiodic [21].

Q: You didn't define that notion.

A: Right, but you can guess: $S = (S_f, S_\omega)$ is aperiodic if the semigroup S_f is.

Q: I suspect that the algebraic characterization of first order logic is not an isolated result.

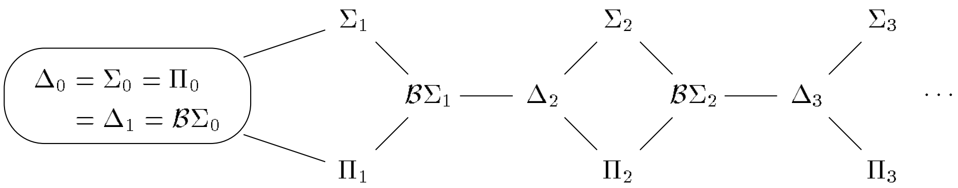
A: No. First, Büchi's theorem also has an algebraic formulation: a language is expressible in the monadic second order logic if and only if it is recognized by a finite semigroup. But there is more. Do you know about Σ_n and Π_n formulæ?

Q: I think so. You write first order formulæ in prenex normal form and then you count the number of alternations between existential and universal quantifiers. A formula is in Σ_n (resp. Π_n) if it starts with a series of existential (universal) quantifiers and has n alternations. For instance, $\exists x_1 \exists x_2 \forall x_3 \forall x_4 \forall x_5 \exists x_6\, \varphi$, where φ is quantifier free, is in Σ_3.

A: Let me introduce two other classes: $\mathrm{B}\Sigma_n$ is the smallest class containing Σ_n and closed under disjunction and negation, and Δ_n is the class of all sentences which are both $*$-equivalent to a Σ_n-sentence and to a Π_n-sentence. The "B" in $\mathrm{B}\Sigma_n$ stands for boolean operations.

Q: It reminds me the arithmetical hierarchy, except that your $*$-equivalence is rather restricted. Do you have similar inclusions?

A: Yes, inclusions are represented in the following diagram. An edge between two classes means that the class on the left is a subclass of the class on the right.



It is also known that the hierarchy Σ_n (respectively Π_n) is proper, if the alphabet contains at least two letters.

Q: I guess you want to know the expressive power of all these classes.

A: Yes, and I would like to know whether the corresponding classes of languages are decidable.

Q: I think I understand better your problem in terms of logic. What you want to do is this: given a first order sentence (or even a monadic second order sentence) φ and an integer n, decide whether φ is $*$-equivalent to a sentence of Σ_n (resp. Π_n, $\mathrm{B}\Sigma_n$, Δ_n.)

A: Right. And you can of course ask the same question for $\mathbb{N}$-equivalence.

Q: Let's stay with finite words first. Yuri told me once that finite structures are very important. Do you have any nice description of the languages corresponding to each level of your hierarchy?

A: Yes. There is a natural hierarchy of star-free languages corresponding to the Σ_n hierarchy. Let me introduce two convenient definitions. The *polynomial closure* of a class of languages $\mathcal{L}$ of A^* is the set of languages that are finite unions of languages of the form $L_0 a_1 L_1 \cdots a_n L_n$, where the a_i's are letters and the L_i's are elements of $\mathcal{L}$. The *boolean closure* of a class of languages $\mathcal{L}$ of A^* is the smallest set of languages containing $\mathcal{L}$ and closed under finite union and complement. By alternating the use of the polynomial closure and of the boolean closure one gets a hierarchy of star-free languages, defined as follows:

(1) $\emptyset$ and A^* are the only languages of level 0

(2) for every integer $n \geq 0$, level $n + 1/2$ is the polynomial closure of level n

(3) for every integer $n \geq 0$, level $n + 1$ is the boolean closure of level $n + 1/2$.

This hierarchy is known as the Straubing hierarchy and is a variation of the "dot-depth hierarchy" introduced by Brzozowski in the sixties. It is interesting to note that this hierarchy was introduced in language theory totally independently from the Σ_n-hierarchy. Now we can state:

(1) A language is $\mathcal{B}\Sigma_n$-definable if and only if it is of level n

(2) A language is Σ_{n+1}-definable if and only if it is of level $n + 1/2$

(3) A language is Π_{n+1}-definable if and only if its complement is of level $n + 1/2$

The connection between the two hierarchies was discovered by Thomas [47]. Actually, Thomas's original result concerned the dot-depth hierarchy and didn't include the half levels, but it is easy to adapt his proof. Another proof and an extension to infinite words are given in [25].

Q: You didn't mention the Δ_n's yet. Do you have any similar result?

A: Yes. Let us say that a product of the form $L = L_0 a_1 L_1 \cdots a_n L_n$ is *unambiguous* if every word u of L admits a unique factorization of the form $u_0 a_1 u_1 \cdots a_n u_n$ with $u_0 \in L_0$, $u_1 \in L_1$, $\ldots$, $u_n \in L_n$. The *unambiguous polynomial closure* of a class of languages $\mathcal{L}$ of A^* is the set of languages that are finite unions of unambiguous products of the form $L_0 a_1 L_1 \cdots a_n L_n$, where the a_i's are letters and the L_i's are elements of $\mathcal{L}$. Pascal Weil and myself proved in [35] that, under certain natural conditions on $\mathcal{L}$, a language L belongs to the unambiguous polynomial closure of $\mathcal{L}$ if and only if L and its complement belong to the polynomial closure of $\mathcal{L}$. In particular:

(4) A language is Δ_{n+1}-definable if and only if it belongs to the unambiguous polynomial closure of the languages of level n.

Q: This reminds me a result of André Arnold [1] in a different context. A set of infinite words is Σ^1_1 (analytic) if and only if it is accepted by a countable Büchi automaton and it is a Borel set if and only if it is accepted by a countable unambiguous Büchi automaton. Now, by Suslin's theorem, $\Sigma^1_1 \cap \Pi^1_1 = \Delta^1_1$ is the class of Borel sets. Thus a set of words is Δ^1_1 if and only if it is accepted by a countable unambiguous Büchi automaton.

A: This is an interesting comparison. By the way I don't know whether the analog of Suslin's separation lemma holds for Σ_n: given two disjoint Σ_n-definable languages L_0 and L_1, does there exist a Δ_n-definable language containing L_0 and disjoint from L_1?

Q: Let me come back to your decidability problems. Are there any algebraic characterizations in the spirit of Schützenberger's theorem for all levels of Straubing's hierarchy?

A: Yes, the $B\Sigma_n$-definable languages can be characterized by a set of identities of their syntactic monoid. An analogous result holds for the Δ_n-definable languages.

Q: What do you mean by "identities"?

A: The precise definition would push us to far afield. Something similar to the identities $x^n = x^{n+1}$ of the aperiodic monoids. For instance, by a deep result of Simon [37], a language is $B\Sigma_1$-definable if and only if its syntactic monoid satisfies the identities $(xy)^n x = (xy)^n = y(xy)^n$ for some n.

Q: What about the Σ_n- and the Π_n-hierarchies?

A: You need a little more than the syntactic monoid. You remember this quasiorder $\leq_L$ I introduced to define the syntactic congruence $\sim_L$?

Q: Yes.

A: By projection, it defines an order relation on the syntactic monoid M of L, called the *syntactic order*. If you prefer, if $\eta : A^* \to M$ is the syntactic morphism of L, and if $P = \eta(L)$, then the syntactic order is defined on M by $x \leq y$ if and only if, for every $u, v \in M$, $uyv \in P$ implies $uxv \in P$. The ordered monoid $(M, \leq)$ is called the *syntactic ordered monoid* of L [30]. Now, the Σ_n-definable (resp. Π_n-definable) languages are characterized by a set of identities of their ordered syntactic monoid [30, 35]. For instance, a language is Σ_1-definable if and only if its syntactic ordered monoid satisfies the identity $x \leq 1$.

Q: Thus all these classes are decidable?

A: Maybe, but this is an open problem, except for the low level classes.

Q: I don't understand. If I have a first order (or monadic second order) sentence φ, I can effectively compute the language $L^*(\varphi)$ and its syntactic monoid. Now I just have to check whether this monoid satisfies the identities of the class I want to study.

A: There are two flaws in your argument. First, it could happen that this set of identities is not recursively enumerable. Second, I never told you that the identities were known for all classes.

Q: But you gave me several examples!

A: The examples I gave you cover almost all what is known. Identities are explicitely known for the classes $B\Sigma_0$, Σ_1, Π_1, $B\Sigma_1$, Δ_2, Σ_2 and Π_2. It follows that these classes are decidable.

Q: What is known about the complexity of the decision problem for these classes?

A: One can decide in polynomial time whether a language of A^* accepted by a deterministic n-state automaton is $B\Sigma_0$- (resp. Σ_1-, Π_1-, $B\Sigma_1$-, Δ_2-) definable. For Σ_2 and Π_2, the best known algorithm is polynomial in $2^{|A|}n$. See [39, 35].

Q: So, if I refer to your diagram, the next step would be $B\Sigma_2$. What is known for this class?

A: Straubing [41] proved that for a two-letter alphabet, the class is decidable. But the general case remains open, and this is actually quite a fascinating problem, which already motivated several articles. Partial results give some evidence that the class should be decidable, but this is still a conjecture. Would you like to know more about it?

Q: Yes, if you don't mind.

A: Depending on your background, you can attack it by combinatorial, algebraic or logical arguments. Let's start with the combinatorial aspects. By definition, the languages of A^* of level 1 (those corresponding to $B\Sigma_1$) are the boolean closure of the languages of the form

$$A^*a_1A^*a_2A^* \cdots A^*a_kA^*$$

where the a_i's are letters. This class is decidable by Simon's theorem. Now the languages of A^* of level 2 (those corresponding to $B\Sigma_2$) are the boolean closure of the languages of the form

$$A_0^*a_1A_1^*a_2A_2^* \cdots A_{k-1}^*a_kA_k^*$$

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

Q: I don't see why.

A: It is not a direct consequence of the definition, but a non trivial result of Howard Straubing and myself [34]. But still, languages of level 2 look very similar to languages of level 1. However, it is not known yet whether the latter class is decidable.

Q: Hmm ...

A: Let's have a look at the algebraic aspects. I don't want to be too technical, but I need a few more definitions. A *submonoid* of a monoid M is a subset of M containing the identity of M and stable under product. A monoid N is a *quotient* of M if there exists a surjective morphism $\varphi : M \to N$. Finally, one says that M *divides* N if M is isomorphic to a quotient of a submonoid of N. I let you verify that division is a transitive relation on monoids. Now let T_n be the set of all upper-triangular boolean square matrices of size n and U_n be the set of matrices of T_n whose diagonal entries are all equal to 1. Then T_n and U_n are monoids under the (boolean) multiplication of boolean matrices.

Q: Let me see. For $n = 3$, typical matrices of T_3 and U_3 look respectively like this, where the ε 's are either 0 or 1.

$$\begin{pmatrix} \varepsilon_{1,1} & \varepsilon_{1,2} & \varepsilon_{1,3} \\ 0 & \varepsilon_{2,2} & \varepsilon_{2,3} \\ 0 & 0 & \varepsilon_{3,3} \end{pmatrix} \qquad \begin{pmatrix} 1 & \varepsilon_{1,2} & \varepsilon_{1,3} \\ 0 & 1 & \varepsilon_{2,3} \\ 0 & 0 & 1 \end{pmatrix}$$

A: Yes. Now one can show that deciding whether a given regular language is of level 1 (resp. 2) amounts to decide whether a given finite monoid divides some U_n (resp. T_n).

Q: It is easy to check whether a given finite monoid divides U_n (resp. T_n) for a fixed n, because a finite monoid only has a finite number of divisors. Thus I guess the problem is to find a bound on n .

A: Exactly. The problem is solved for U_n but still open for T_n.

Q: Both problems look very close yet.

A: Yes, and this is quite frustrating to have the solution only for one of them.

Q: What about the logical approach? Do you think my logician friends could help?

A: Very likely. I already mentionned the back and forth arguments (Fraïssé-Ehrenfeucht games) used by Thomas and Wilke. Christian Glaßer and Heinz Schmitz recently obtained some very nice results [15] by using ideas issued from descriptive set theory. If you have look at their preprints on their web page, you will see that their forthcoming papers are quite promising.

Q: Do you know anything for the upper levels?

A: It is known, in a rather precise way [35], how to find the identities defining level $n + 1/2$ (or Σ_n if you prefer) given the identities defining the level n (or $\mathrm{B}\Sigma_{n-1}$). I can only give you a flavour of this result.

Q: Fine.

A: You know about kernels in group theory. When you have a group morphism $\varphi : G \to H$ between two groups G and H, the set $Ker(\varphi) = \varphi^{-1}(1)$ is a normal subgroup of G. When you have a monoid morphism $\varphi : M \to N$, you want to consider not only the inverse image of the identity, but also the inverse image $\varphi^{-1}(e)$ of each idempotent e of N.

Q: An idempotent is an element e such that $e^2 = e$, right?

A: Yes. You remember that the languages of the full (resp. half) levels are characterized by their (resp. ordered) syntactic monoid. So we may speak freely of finite (ordered) monoids of level n (resp. $n + 1/2$). If you have an ordered monoid $(M, \leq)$, a monoid N and a monoid morphism $\varphi : M \to N$, then $\varphi^{-1}(e)$ is an ordered subsemigroup of M for every idempotent e of N. Now here is the key result: if N is of level n and if, for all idempotent $e \in N$, $\varphi^{-1}(e)$ satisfies the identity $x^k y x^k \leq x^k$ for some k, then $(M, \leq)$ is of level $n + 1/2$.

Q: So if you expand a monoid of level n by ordered semigroups satisfying $x^k y x^k \leq x^k$ for some k, you get an ordered monoid of level $n + 1/2$.

A: Yes.

Q: And to pass from level $n + 1/2$ to level $n + 1$?

A: There are some results and some conjectures, too. But it's definitely too technical for this conversation.

Q: What happens for infinite words?

A: The decidability results known for finite words can be extended to infinite words with the proper definitions. In particular, the same decidability results hold. But, as you have seen before, you need more sophisticated tools to deal with infinite words.

Q: Does the algebraic approach apply to other cases?

A: Straubing, Thérien and Thomas [43] gave a syntactic characterization of first order logic with generalized quantifiers. The logic in which the relation $<$ is replaced by the successor relation was also considered by Thomas [47]. In this case the first order definable languages form a strict subclass of the star-free languages which also has a characterization by identities, originally discovered by Thérien and Weiss [3, 4, 44, 50], but the Σ_n-hierarchy collapses at the second level. The decidability of the classes Σ_1 and $\mathrm{B}\Sigma_1$ was also recently established [50, 31]. Blanchet-Sadri investigated the connections between some refinements of the dot-depth hierarchy and Fraïssé-Ehrenfeucht games [5, 6, 7].

Another application of this approach concerns propositional linear temporal logic, interpreted on finite and infinite words. A well known result of Kamp [16, 14] states that this logic has the same expressive power as the first order logic we talked about. So a language (resp. an ω-language) is expressible in linear temporal logic if and only if it is star-free. This result can be proved directly by algebraic methods [11, 12]. Restricted temporal logics may also be considered.

Q: What do you mean by restricted temporal logic?

A: Consider for instance the "temporal logic without Until", in which you are just allowed to use the temporal operators Next and Eventually. Its expressive power on finite words can be specified by identities [12]. It follows that one can decide whether a given temporal formula is $*$-equivalent with a formula without Until. Recently, Thomas Wilke gave a thoroughly study of such fragments of temporal logic [51]. They are all decidable, for finite words and for infinite words.

Q: I am a bit tired, and I need to assimilate all what you said. What would you suggest me to read?

A: There are several survey papers you could read [24, 26, 32, 29, 38, 48]. Then you can compulse the references given in these papers to go further on.

Q: Before I go, why did you get interested into logic?

A: Because I enjoyed the results of the paper of Wolfgang Thomas [47], but at the time, I was not able to understand the proof! But there is also something more, how to say, æsthetic. I have the feeling that there is something deep in this connection between the most important family of regular languages — the star-free languages —, the most important class of finite monoids — the aperiodic monoids — and the most important fragment of logic — the first order logic —.

Q: I see what you mean. It's a good conclusion for our conversation. Thank you.

Acknowledgements

I would like to thank Yuri Gurevich for inviting me to have this conversation with his student Quisani. I am also very grateful to Kevin Compton who gave me the opportunity to meet Quisani and to Juris Hartmanis for an enthusiastic ten minute conversation on this topic in India.

References

[1] A. Arnold, 1983, Topological characterizations of infinite behaviours of transition systems, in *Automata, Languages and Programming, J. Diaz ed., Lecture Notes in Computer Science* **154**, 28–38, Springer.

[2] A. Arnold, 1985, A syntactic congruence for rational ω-languages, *Theoret. Comput. Sci.* **39**, 333–335.

[3] D. Beauquier and J.-E. Pin, 1989, 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, 63–79.

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

[5] F. Blanchet-Sadri, 1989, Games, equations and the dot-depth hierarchy, *Comput. Math. Appl.* **18**, 809–822.

[6] F. Blanchet-Sadri, 1992, Games, equations and dot-depth two monoids, *Discrete Appl. Math.*, **39**, 99–111.

[7] F. Blanchet-Sadri, 1995, Some logical characterizations of the dot-depth hierarchy and applications, *J. Comput. System Sci.* **51**, 324–337.

[8] J.R. Büchi, 1960, Weak second-order arithmetic and finite automata, *Z. Math. Logik und Grundl. Math.* **6**, 66–92.

[9] J.R. Büchi, 1962, 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, 1–11.

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

[11] J. Cohen, 1991, On the expressive power of temporal logic for infinite words, *Theoret. Comput. Sci.* **83**, 301–312.

[12] J. Cohen, D. Perrin and J.-E. Pin, 1993, On the expressive power of temporal logic, *J. Comput. System Sci.* **46**, 271–294.

[13] K. Compton and H. Straubing, 1992, Characterizations of Regular Languages in Low-Level Complexity Classes, the Logic in Computer Science Column, Bull. EATCS **48**, 134–142.

[14] D. Gabbay, A. Pnueli, S. Shelah and J. Stavi, 1980, On the temporal analysis of fairness, *Proc. 7th ACM Symp. Princ. Prog. Lang.*, 163–173.

[15] C. Glaßer and H. Schmitz, 2000, Languages of dot-depth 3/2, *Proceedings 17th Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Comput. Sci. **1770**, Springer, 555–566.

[16] J.A. Kamp, 1968, Tense logic and the theory of linear order, Ph. D. Thesis, Univ. of California, Los Angeles.

[17] R. Ladner, 1977, Application of model theoretic games to discrete linear orders and finite automata, *Information and Control* **33**, 281–303.

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

[19] J.P. Pécuchet, 1986, Variétés de semigroupes et mots infinis, in B. Monien and G. Vidal-Naquet eds., *STACS 86, Lecture Notes in Comput. Sci.* **210**, Springer, 180–191.

[20] J.P. Pécuchet, 1986, Etude syntaxique des parties reconnaissables de mots infinis, in *Proc. 13th ICALP, (L. Kott ed.) Lecture Notes in Comput. Sci.* **226**, 294–303.

[21] D. Perrin, 1982, Variétés de semigroupes et mots infinis, *C.R. Acad. Sci. Paris* **295**, 595–598.

[22] D. Perrin, 1984, Recent results on automata and infinite words, in *Mathematical Foundations of Computer Science*, Lecture Notes in Comput. Sci. **176** Springer-Verlag, New-York/Berlin, 134–148.

[23] D. Perrin, 1984, An introduction to automata on infinite words, in *Automata on Infinite Words* (Nivat, M. ed.), Lecture Notes in Comput. Sci. **192**, Springer, 2–17.

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

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

[26] D. Perrin and J.-E. Pin, 1995, Semigroups and automata on infinite words, in NATO Advanced Study Institute *Semigroups, Formal Languages and Groups, J. Fountain (ed.)*, Kluwer academic publishers, 49–72.

[27] D. Perrin and J.-E. Pin, *Infinite words*, to appear.

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

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

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

[31] J.-E. Pin, 1996, The expressive power of existential first order sentences of Büchi's sequential calculus, *ICALP 1996, Lecture Notes in Comput. Sci.* **1099**, Springer, 300–311.

[32] J.-E. Pin, 1996, Logic, Semigroups and Automata on Words, *Annals of Mathematics and Artificial Intelligence* **16**, 343–384.

[33] J.-E. Pin, 1997, Syntactic semigroups, Chapter 10 in *Handbook of formal languages*, G. Rozenberg and A. Salomaa eds., Springer.

[34] J.-E. Pin and H. Straubing, 1981, Monoids of upper triangular matrices, *Colloquia Mathematica Societatis Janos Bolyai 39, Semigroups, Szeged*, 259–272.

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

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

[37] I. Simon, 1975, Piecewise testable events, *Proc. 2nd GI Conf., Lect. Notes in Comp. Sci.* **33**, Springer, Berlin, 214–222.

[38] I. Simon, 1993, The product of rational languages, *Proceedings of ICALP 1993, Lecture Notes in Computer Science* **700**, 430–444.

[39] J. Stern, 1985, Complexity of some problems from the theory of automata, *Inform. and Control* **66**, 163–176.

[40] L. Stockmeyer, 1977, The polynomial time hierarchy, *Theoret. Comp. Sci.* **3**, 1–22.

[41] H. Straubing, 1988, Semigroups and languages of dot-depth two, *Theoret. Comp. Sci.* **58**, 361–378.

[42] H. Straubing, 1994, Finite Automata, Formal Logic, and Circuit Complexity, Birkhäuser, Boston.

[43] H. Straubing, D. Thérien and W. Thomas, 1988, Regular Languages Defined with Generalized Quantifiers, in *Proc. 15th ICALP*, Springer Lecture Notes in Computer Science **317**, 561–5 75.

[44] D. Thérien and A. Weiss, 1985, Graph congruences and wreath products, *J. Pure Applied Algebra* **35**, 205-215.

[45] W. Thomas, 1979, Star-free regular sets of ω-sequences, *Information and Control* **42** 148–156.

[46] W. Thomas, 1981, A combinatorial approach to the theory of ω-automata, *Information and Control* **48**, 261–283.

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

[48] W. Thomas, 1990, Automata on infinite objects, in *Handbook of Theoretical Computer Science, vol B, Formal models and semantics*, Elsevier, 135–191.

[49] T. Wilke, 1993, An algebraic theory for regular languages of finite and infinite words, *International Journal of Algebra and Computation* **3**, 447–489.

[50] T. Wilke, 1993, 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, 607–616.

[51] T. Wilke, 1999, Classifying discrete temporal properties, in *STACS 99, Christoph Meinel and Sophie Tison (Eds.), Lect. Notes in Comp. Sci.* **1563**, Springer, Berlin, 32- -46.