Over Words, Two Variables Are as Powerful as One Quantifier Alternation: $\mathrm{FO}^2 = \Sigma_2 \cap \Pi_2$

Denis Thérien
School of Computer Science, McGill University
3480 University Montréal, Québec H3A 2A7, Canada
denis@opus.cs.mcgill.ca

Thomas Wilke
Institut für Informatik und Praktische Mathematik
Christian-Albrechts-Universität zu Kiel
24098 Kiel, Germany
tw@informatik.uni-kiel.de

Abstract. We show a property of strings is expressible in the two-variable fragment of first-order logic if and only if it is expressible by both a Σ_2 and a Π_2 sentence. We thereby establish:

$$\mathrm{UTL} = \mathrm{FO}^2 = \Sigma_2 \cap \Pi_2 = \mathrm{UL} \ ,$$

where UTL stands for the string properties expressible in the temporal logic with 'eventually in the future' and 'eventually in the past' as the only temporal operators and UL stands for the class of unambiguous languages. This enables us to show that the problem of determining whether or not a given temporal string property belongs to UTL is decidable (in exponential space), which settles a hitherto open problem.

Our proof of $\Sigma_2 \cap \Pi_2 = \mathrm{FO}^2$ involves a new combinatorial characterization of these two classes and introduces a new method of playing Ehrenfeucht-Fraïssé games to verify identities in semigroups.

While the number of variables required to express a certain *graph* property in first-order logic is an important measure for the descriptional complexity of the property in question [Imm91], the measure to classify properties of *strings* that has received most attention is the number of quantifier alternations. In fact, the number of quantifier alternations induces a strict hierarchy on the class of all first-order expressible string properties which is identical with the dot-depth hierarchy [CB71, BK78, Tho82, PP86], the natural hierarchy on the star-free sets. The reason the number of variables has played a minor role in the realm of strings is that three variables are sufficient to express any first-order expressible string property [Kam68, IK89]. Only recently, it was, however, shown that even in the restricted setting of strings the number of variables used is a relevant parameter [EVW97]: the properties expressible with two variables are exactly the properties expressible in temporal logic without 'until' and 'since', a fragment of temporal logic also known as unary temporal logic.

In the present paper, we reveal a neat and natural connection between two variables and quantifier alternation, namely, we show a property of strings is expressible with two variables if and only if it is expressible by both a $\forall^*\exists^*$ sentence and an $\exists^*\forall^*$ sentence. The class of all such properties is identical to the class of unambiguous languages [PW97], a well-studied class of formal languages which admits a characterization in terms of semigroups just as the class of star-free languages [Sch76]: a regular language is unambiguous if and only if its syntactic semigroup belongs to **DA**, a specific subclass of all "group-free" (aperiodic) semigroups.

We give a new combinatorial characterization of the semigroups in **DA** in terms of a parameterized family of congruence relations and prove that these congruence relations capture exactly the properties expressible with two variables. In order to show that every such property is an unambiguous language, we use Ehrenfeucht-Fraïssé games in a new fashion; we play these games to verify that the equation defining **DA** holds in the syntactic semigroups of all string properties expressible with two variables.

The decidability of the class of all unambiguous languages [Sch76] implies that the problem of determining whether or not a temporal string property is expressible in unary temporal logic, and thus solves an open problem from [EVW97]. We show the problem is decidable in exponential space.

When properties of strings are expressed in such weak fragments of first-order logic as the two-variable fragment it makes a difference whether or not the "successor" predicate is available (in addition to the built-in predicate "less than"). We prove that $\mathrm{FO}^2 = \Sigma_2 \cap \Pi_2$ holds true in both scenarios.

The identity $\mathrm{FO}^2 = \Sigma_2 \cap \Pi_2$ does not carry over to infinite words,[1] but using the results about strings we can nevertheless show that it is decidable whether or not a regular ω-language is expressible in unary temporal logic or $\Sigma_2 \cap \Pi_2$.

The paper is organized as follows. In Section 1, we review the terminology from logic, formal language theory, and semigroup theory we need and state the main result of the paper. Section 2 presents our new combinatorial characterization, and Sections 3 and 4 are devoted to the proof that our characterization is correct. Section 5 describes how our results carry over to the situation where successor is available.

[1] Consider the property denoted by the regular expression $(a+b)^*(ab^*)^\omega$, which is not Σ_2 expressible.

1 Different Characterizations of the Same String Properties

A *property of strings* is just a set of non-empty strings over a given finite alphabet, that is, a formal language. For simplicity, we assume an infinite supply of letters, $a_0, a_1, a_2, \ldots$, and consider only alphabets of the form $\{a_0, \ldots, a_{m-1}\}$, denoted A_m. A string u of length n over A_m is viewed as a relational structure $(\{0, \ldots, n-1\}, <^n, P_0^u, \ldots, P_{m-1}^u)$ where P_i^u contains all positions in u labeled a_i and $<^n$ is the usual built-in "less than" predicate. Accordingly, our first-order vocabulary is $\{<, P_0, \ldots, P_{m-1}\}$, which we denote by σ_m.

We investigate the power of two different fragments of the first-order logics over the vocabularies σ_m. Firstly, we are interested in the class of string properties that are expressible by a sentence with at most two variables, a class of string properties denoted by $\mathrm{FO}^2[<]$. Secondly, we are interested in the class of string properties that are expressible, at the same time, by a sentence of the form $\exists x_0 \ldots \exists x_{k-1} \forall y_0 \ldots \forall y_{l-1} \phi$ and a sentence of the form $\forall x_0 \ldots \forall x_{k-1} \exists y_0 \ldots \exists y_{l-1} \phi$ where in both cases ϕ is quantifier-free and k and l are arbitrary natural numbers. This class is denoted by $\Sigma_2 \cap \Pi_2[<]$.

For $\mathrm{FO}^2[<]$, the following characterization was obtained only recently.

Theorem 1 ([EVW97]) $\mathrm{FO}^2[<] = \mathrm{TL}[\diamondplus\!\!\!\!\diamondminus]$.

Here, $\mathrm{TL}[\diamondplus\!\!\!\!\diamondminus]$ stands for the class of properties expressible in the fragment of temporal logic that uses "eventually in the future" ($\diamondplus$) and "eventually in the past" ($\diamondminus$) as the only temporal operators.

For $\Sigma_2 \cap \Pi_2[<]$, there are two important characterizations:

Theorem 2 ([Sch76], [PW97]) $\Sigma_2 \cap \Pi_2[<] = \mathrm{UL} = \mathrm{DA}$.

Here, UL stands for the class of all *unambiguous languages*, which we describe now in more detail. A product of the form $B_0^* a_1 B_1^* a_2 \ldots a_k B_k^*$ with $B_i \subseteq A$ and $a_i \in A$ is said to be *unambiguous* if every string allows at most one factorization, i. e., if for each string u there is at most one sequence $(u_0, u_1, \ldots, u_k)$ such that $u = u_0 a_1 u_1 a_2 \ldots a_k u_k$ and $u_i \in B_i^*$ for $i \leq k$. A language is *unambiguous* if it is a finite disjoint union of unambiguous products.

Further, DA stands for the class of regular languages whose syntactic semigroups belong to $\mathbf{DA}$, the class of finite semigroups satisfying

$$(xyz)^\omega y (xyz)^\omega = (xyz)^\omega \quad . \tag{1}$$

In this identity, $^\omega$ denotes the operator that maps every element s of a finite semigroup to the unique element in $\{s, s^2, s^3, \ldots\}$ which is idempotent, i. e., which satisfies $x^2 = x$.

As a consequence of the above theorem it is decidable whether or not a given regular language (say, represented by a finite automaton, a temporal formula, a first-order sentence, or even a monadic second-order sentence) belongs to $\Sigma_2 \cap \Pi_2[<]$. This is because one can effectively compute the syntactic semigroup (its multiplication table) of a regular language and, of course, check whether an identity such as (1) holds or not.

The main result of our paper is that all the five classes of string properties defined so far are in fact identical:

Theorem 3 $\mathrm{FO}^2[<] = \Sigma_2 \cap \Pi_2[<] = \mathrm{TL}[\diamondplus] = \mathrm{UL} = \mathrm{DA}$.

This result, together with the characterization of **DA** via (1), enables us to show:

Theorem 4 *The problem of determining whether or not a string property represented by a deterministic finite automaton belongs to one of the classes* $\mathrm{FO}^2[<]$ *or* $\mathrm{TL}[\diamondplus]$ *(or* $\Sigma_2 \cap \Pi_2[<]$, UL, DA*) is in* **PSPACE**.

This means, in particular, that when we are given an arbitrary temporal formula we can decide in exponential space[2] whether or not the formula is equivalent (over strings) to a formula with "eventually in the future" and "eventually in the past" as the only temporal operators; in [EVW97], this decidability question was left open.

2 The Combinatorial Description

In order to prove Theorem 3, we introduce yet another class of string properties, which we show is identical with $\mathrm{FO}^2[<]$ and DA, and thus works as a link between the two worlds from Theorems 1 and 2. This class is defined via a parameterized family of equivalence relations on strings which we describe in the following.

Let u be an arbitrary string. Write $\alpha(u)$ for the set of letters occurring in u. Assume $a \in \alpha(u)$. Then there is a unique triple (u_0, a, u_1) such that $u = u_0 a u_1$ and $a \notin \alpha(u_0)$, which we call the *a-left decomposition* of u. Symmetrically, there is a unique triple (u_0, a, u_1) such that $u = u_0 a u_1$ and $a \notin \alpha(u_1)$, which we call the *a-right decomposition* of u.

For each natural number n and strings u and v, we define whether or not $u \equiv_n v$, by induction on $n + |\alpha(u)|$.

First, we define that $u \equiv_0 v$ holds for all strings u and v. Second, if $n > 0$, then $u \equiv_n v$ if and only if

[2] The conversion of a temporal formula into an equivalent finite automaton incurs an additional blowup.

1. $\alpha(u) = \alpha(v)$, and
2. for every $a \in \alpha(u)$, if (u_0, a, u_1) and (v_0, a, v_1) are the a-left decompositions of u and v, then $u_0 \equiv_n v_0$ and $u_1 \equiv_{n-1} v_1$, and
3. for every $a \in \alpha(u)$, if (u_0, a, u_1) and (v_0, a, v_1) are the a-right decompositions of u and v, then $u_0 \equiv_{n-1} v_0$ and $u_1 \equiv_n v_1$.

(Note that this is well-defined, since $|\alpha(u_0)| < |\alpha(u)|$ in (2) and $|\alpha(u_1)| < |\alpha(u)|$ in (3).)

So, the $\equiv_1$ class of a string u tells us, for instance, what letters occur in u and how the first occurrences of these letters and their last occurrences are ordered. Moreover, it tells us how these orders are for the prefixes of u and suffixes of u that extend to a first respectively last occurrence of a letter, and so on. In general, the relations $\equiv_n$ are defined in such a way that for growing n one obtains an increasing amount of information about the order of first and last occurrences of letters and recursively so about segments that are delimited by first and last occurrences of letters. And—that is very important—the relations $\equiv_n$ are defined in such a way that they are congruence relations:

Lemma 1 *For every $n \geq 0$, the relation $\equiv_n$ is a congruence relation (i.e., $uu' \equiv_n vv'$ whenever $u \equiv_n v$ and $u' \equiv_n v'$).*

We write $\equiv_n^m$ for the restriction of $\equiv_n$ to non-empty strings over A_m (that is, $\equiv_n^m$ is $\equiv_n \cap (A_m^+ \times A_m^+)$) and say that a language $L \subseteq A_m^+$ is *n-testable* if it is a union of classes of $\equiv_n^m$.

What we prove is the following, which, in view of Theorems 1 and 2, implies Theorem 3.

Theorem 5 (combinatorial characterization) *Let $L \subseteq A_m^+$, for some $m \geq 0$. Then the following assertions are equivalent.*

1. *$L \in \mathrm{FO}^2[<]$.*
2. *The syntactic semigroup of L belongs to* **DA**.
3. *L is n-testable for some $n \geq 0$.*

In the following two sections, we will sketch the proofs of this theorem; the implications (1) $\Rightarrow$ (2) and (3) $\Rightarrow$ (1) are dealt with in Section 3, and (2) $\Rightarrow$ (3) is proved in Section 4.

3 The Game-Theoretic Arguments

We make extensive use of the 2-pebble Ehrenfeucht-Fraïssé game (2EFG) [Imm82], which we recall here briefly. In our context, such a game is played on two strings by two players, called Spoiler (male) and Duplicator (female). Altogether, there are four pebbles, two blue and two red pebbles. With either string, two pebbles of different colors are associated. At the beginning of the game, the four pebbles sit outside the strings. The game proceeds in rounds, and in each round Spoiler picks up one pebble and puts it on a position of the associated string. Duplicator is then required to put the other pebble of the same color on a position of the other string in such a way that first, positions pebbled with the same color carry the same letter and second, the order in which the blue and red pebble occur on the strings is the same. If Duplicator cannot put the pebble such that this requirement is met, she looses; if in an r-round game Duplicator can answer correctly in each of the r rounds, she wins.

We write $u \cong_r^m v$ when u and v are non-empty strings over A_m and Duplicator has a winning strategy in the r-round 2EFG on u and v. The fundamental result about these games is: a language $L \subseteq A_m^+$ belongs to $\mathrm{FO}^2[<]$ if and only if there exists a number r such that L is a union of classes of $\cong_r^m$.

3.1 The First Game-Theoretic Argument

It is easily seen that, for every $m, r \geq 0$, the equivalence relation $\cong_r^m$ is a congruence relation. Therefore, in order to prove that (1) implies (2) in Theorem 5 it is enough to show:

Proposition 1 *For every $m, r \geq 0$, the semigroup $A_m^+/\cong_r^m$ satisfies (1).*

In the following, we sketch the proof of this proposition. Fix r and m. For the claim to be true, it is sufficient that there exists a number n_0 such that for all $n \geq n_0$ and all $x_1, x_2, x_3 \in A_m^+$ Duplicator wins the r-round 2EFG on $(x_1x_2x_3)^n x_2 (x_1x_2x_3)^n$ and $(x_1x_2x_3)^{2n}$. Clearly, without loss of generality, we can assume that x_1, x_2, and x_3 are distinct letters, say $x_1 = a$, $x_2 = b$, and $x_3 = c$.

We claim that for all $n > r$ Duplicator wins the r-round 2EFG on $(abc)^n b (abc)^n$ and $(abc)^{2n}$ and give an informal description of Duplicator's winning strategy.

Let $n > r$ and set $u = (abc)^n b (abc)^n$ and $v = (abc)^{2n}$. There is a natural correspondence between the first and last $3n$ positions in u on the one hand and the positions in v on the other hand. When it is Duplicator's turn, she always tries to pebble the corresponding position in the other string. She cannot do so, when Spoiler pebbles the middle position of u, or might not be able to do so, when the game is already in a configuration where the pebbles are on positions which do not correspond. In the first case, she will pebble one of the two positions next to the center of v labeled b. In the second case, she will pebble a position which is next to the corresponding position and has the same label. One can show that this strategy always works; the important observation is that after r' rounds played, the pebbles are in corresponding positions or they are at least $3(n - r')$ positions away from either end of the string they are on.

3.2 The Second Game-Theoretic Argument

In view of what we said about the 2EFG at the beginning of this section, it is clear that the following proposition implies that every n-testable language belongs to $\mathrm{FO}^2[<]$, the implication from (3) to (1) in Theorem 5.

Proposition 2 *Assume $u \not\equiv_n v$ for some strings u and v. Then Spoiler wins the $(n + |\alpha(u)|)$-round 2EFG on u and v.*

The proof goes by induction on $n + \alpha(u)$. If $n = 0$ there is nothing to show, as all strings are 0-equivalent. So suppose $n > 0$.

First, assume $\alpha(u) \neq \alpha(v)$. Then Spoiler wins easily within one round; he just pebbles a position which is labeled with a letter that does not occur in the other string. Second, assume $\alpha(u) = \alpha(v)$. Then there is some $a \in \alpha(u)$ such that (without loss of generality) if (u_0, a, u_1) and (v_0, a, v_1) are the a-left decompositions of u and v, then $u_0 \not\equiv_n v_0$ or $u_1 \not\equiv_{n-1} v_1$. We only deal with the first case, the other is similar. So assume $u_0 \not\equiv_n v_0$. By induction hypothesis, Spoiler has a winning strategy in the $(n+|\alpha(u)|-1)$-round game on u_0 and v_0. In the game on u and v, Spoiler simply mimics this strategy on u and v and either wins or at some point Duplicator replies by pebbling a position which is not in u_0 or v_0, say a position i on u such that $i > |u_0|$ (equality is impossible, since position $|u_0|$ is labeled a, but there is no a in v_0). In the next round, Spoiler puts the other pebble on i in u and wins, because Duplicator is forced to reply on a position in v_0 (a position which is to the left of the other pebble on v), but a does not occur in v_0.

4 The Algebraic Argument

That every language whose syntactic semigroup belongs to $\mathbf{DA}$ is n-testable for some n follows from the following proposition, which is what we actually prove.

Proposition 3 *Let $h\colon A^+ \to S$ be a homomorphism into a member of* $\mathbf{DA}$. *Then there exists a number n such that for $u, v \in A^+$, if $u \equiv_n v$, then $h(u) = h(v)$.*

Recall that the relations $\equiv_n$ were defined in such a way that they determine the order of first and last occurrences of letters and recursively of the segments that are delimited by these occurrences (the factors of the a-left and a-right decompositions). We will now explain why this information is useful in the context of **DA**.

Recall that given a semigroup S and $p, q \in S$, one writes $p \leq_{\mathcal{R}} q$, if $\{p\} \cup \{pt \mid t \in S\} \subseteq \{q\} \cup \{qt \mid t \in S\}$, and ones writes $p \mathrel{\mathcal{R}} q$, if $p \leq_{\mathcal{R}} q$ and $q \leq_{\mathcal{R}} p$. The relation $\mathcal{L}$ is defined symmetrically. Recall also that if S is an aperiodic semigroup (in particular, if $S \in \mathbf{DA}$), then $p = q$ whenever $p, q \in S$ are such that $p \mathrel{\mathcal{R}} q$ and $p \mathrel{\mathcal{L}} q$.

Let $h: A^+ \to S$ be a homomorphism and $u \in A^+$. We define the $\mathcal{R}$ *decomposition* of the string u with respect to h to be the unique sequence $(u_0, a_1, u_1, a_2, \ldots, a_l, u_l)$ satisfying:

1. $a_i \in A$ for $1 \leq i \leq l$, and $u_i \in A^*$ for $i \leq l$,
2. $u = u_0 a_1 u_1 \ldots u_{l-1} a_l u_l$,
3. $h(u_0 a_1 u_1 \ldots u_s) \not\mathrel{\mathcal{R}} h(u_0 a_1 u_1 \ldots u_s a_{s+1})$ for $s < l$, and
4. $h(a_0) \mathrel{\mathcal{R}} h(u_0)$ where a_0 is the first letter of u, and $h(u_0 a_1 u_1 \ldots a_s) \mathrel{\mathcal{R}} h(u_0 a_1 u_1 \ldots a_s u_s)$ for $1 \leq s \leq l$.

We prove the following lemma, which connects $\mathcal{R}$ decompositions and iterated a-left decompositions.

Lemma 2 *Let $h: A^+ \to S$ be a homomorphism into a member of* **DA** *and $u \in A^+$. Assume $(u_0, a_1, u_1, a_2, \ldots, a_l, u_l)$ is the $\mathcal{R}$ decomposition of u with respect to h. Then $a_{s+1} \notin \alpha(u_s)$ for $s < l$.*

This says, for instance, that if $(u_0, a_1, u_1, a_2, \ldots, a_l, u_l)$ is the $\mathcal{R}$ decomposition of u, then $(u_0, a_1, u_1 a_2 \ldots a_l u_l)$ is the a_1-left decomposition of u, and, similarly, $(u_1, a_2, u_3 a_3 \ldots a_l u_l)$ is the a_2-left decomposition of the right factor of the a_1-left decomposition of u.

Using this, we can now prove Proposition 3. We choose $n = |A||S| + 1$ and show that if $u \equiv_k v$ for $k > |\alpha(u)||S|$, then $h(u) = h(v)$ (which is clearly enough). The proof of this refined claim goes by induction on $|\alpha(u)|$.

If $|\alpha(u)| = 0$, then $u = v = \varepsilon$ and the claim is trivially satisfied. For the inductive step, assume $|\alpha(u)| > 0$, $k > |\alpha(u)||S|$, and $u \equiv_k v$. Then $\alpha(u) = \alpha(v) \neq \emptyset$. Let $(u_0, a_1, u_1, \ldots, a_l, u_l)$ be the $\mathcal{R}$ decomposition of u with respect to h. Then $l \leq |S|$ (because the number of $\mathcal{R}$ classes is bounded by $|S|$). Write x_i for $u_i a_{i+1} u_{i+1} \ldots u_l$ where $i \leq l$. From Lemma 2 follows that (u_i, a_{i+1}, x_{i+1}) is the a_{i+1}-left decomposition of x_i, for $i < l$. Thus there exists a decomposition $(v_0, a_1, \ldots, v_l)$ of v such that $u_i \equiv_{k-i} v_i$ for $i < l$ where (v_i, a_{i+1}, y_{i+1}) is the a_{i+1}-left decomposition of y_i when we set $y_i = v_i a_{i+1} \ldots v_l$. (Observe that $k - i > (|\alpha(u)| - 1)|S|$ for $i < l$.) By induction hypothesis, we get $h(u_i) = h(v_i)$ for $i < l$. Therefore, $h(u) \mathrel{\mathcal{R}} h(u_0 \dots a_l) = h(v_0 \dots a_l) \geq_{\mathcal{R}} h(v)$. By symmetry, we get $h(v) \geq_{\mathcal{R}} h(u)$, which implies $h(u) \mathrel{\mathcal{R}} h(v)$. Thus, by left-right symmetry, $h(u) \mathrel{\mathcal{L}} h(v)$, hence $h(u) = h(v)$ by the above remark about aperiodic semigroups.

5 Successor Available

Consider the property denoted by the regular expression $(a_0a_1)^+$. This property does not belong to $\mathrm{FO}^2[<]$ (because $(a_0a_1)^n a_0 (a_0a_1)^n \notin L$, but $(a_0a_1)^n \in L$ for every $n > 0$; see Subsection 3.1), but it can very easily be expressed using two variables when in addition to the built-in predicate $<$ the built-in predicate "successor of" is available.

We denote the "successor of" predicate by suc and write σ_m^{suc} for the extension of σ_m by suc. Accordingly, we write $\mathrm{FO}^2[\mathrm{suc}, <]$ for the string properties expressible by a sentence in the two-variable fragment of the first-order logic over σ_m^{suc} for some m. In the same spirit, we use the notation $\Sigma_2 \cap \Pi_2[\mathrm{suc}, <]$. The temporal logic that results from $\mathrm{TL}[\diamondplus]$ by adding "previously" ($\ominus$) and "next" ($\oplus$) is denoted by $\mathrm{TL}[\textcircled{$\pm$}, \diamondplus]$.

In [EVW97], it was shown that $\mathrm{TL}[\textcircled{$\pm$}, \diamondplus]$ and $\mathrm{FO}^2[\mathrm{suc}, <]$ are identical. We extend this result as follows (cf. Theorem 3).

Theorem 6 $\mathrm{FO}^2[\mathrm{suc}, <] = \Sigma_2 \cap \Pi_2[\mathrm{suc}, <] = \mathrm{TL}[\textcircled{$\pm$}, \diamondplus] = \mathrm{DA} * \mathrm{D}$.

Here, $\mathrm{DA} * \mathrm{D}$ stands for the class of languages whose syntactic semigroups belong to the class of finite semigroups denoted by $\mathbf{DA} * \mathbf{D}$. We use a result from [Alm96] about this class to show:

Theorem 7 *The problem of determining whether or not a string property represented by a deterministic finite automaton belongs to one of the classes* $\mathrm{FO}^2[\mathrm{suc}, <]$ *or* $\mathrm{TL}[\textcircled{$\pm$}, \diamondplus]$ *(or* $\Sigma_2 \cap \Pi_2[\mathrm{suc}, <]$*,* $\mathrm{DA} * \mathrm{D}$*) is decidable in* **PSPACE**.

This answers the other version of the open problem from [EVW97].

The idea of the proof of Theorem 6 is based on a reduction to the situation without successor. Let $k > 0$ and u an arbitrary string over an alphabet A. We define a string u^k over $A^{\leq k}$; we let $u^k(i) = (u(0), u(1), \dots, u(i))$ when $i < k$ and $u^k(i) = (u(i-k+1), \dots, u(i))$ otherwise. We then prove the following.

Proposition 4 *Let L be a property of strings over an alphabet A. Then* $L \in \mathrm{FO}^2[\mathrm{suc}, <]$ *(respectively* $\Sigma_2 \cap \Pi_2[\mathrm{suc}, <]$*,* $\mathrm{TL}[\textcircled{$\pm$}, \diamondplus]$*) if and only if there exists a number k and* $L' \in \mathrm{FO}^2[<]$ *(respectively* $\Sigma_2 \cap \Pi_2[<]$*,* $\mathrm{TL}[\textcircled{$\pm$}, \diamondplus]$*) such that $u \in L$ iff $u^k \in L'$, for every $u \in A^+$.*

The connection to DA $*$ D is established via the wreath product principle [Str78].

Conclusion and Open Problem

We have demonstrated how algebraic and model-theoretic concepts together can be used to obtain expressibility as well as algorithmic results about fragments of first-order and temporal logic. Unary temporal logic, which we now know how to decide, is level 0 of the combined until/since hierarchy [EW96]. It would be interesting to see if our methods can be extended to show the decidability of the other levels too; note that the theory developed in [TW96] does not seem to be applicable.

Acknowledgment. We thank Kousha Etessami, Pierre McKenzie, and Oliver Matz for comments on the first draft of this paper.

References

[Alm96] J. Almeida. A syntactical proof of locality of **DA**. *Internat. J. Algebra and Comput.*, 6(2):165–177, 1996.

[BK78] J. A. Brzozowski and R. Knast. The dot-depth hierarchy of star-free languages is infinite. *J. Comput. System Sci.*, 16:37–55, 1978.

[CB71] R. S. Cohen and J. A. Brzozowski. Dot-depth of star-free events. *J. Comput. System Sci.*, 5:1–16, 1971.

[EVW97] K. Etessami, M. Y. Vardi, and Th. Wilke. First-order logic with two variables and unary temporal logic. LICS '97, 228–235.

[EW96] K. Etessami and Th. Wilke. An until hierarchy for temporal logic. LICS '96, 108–117.

[IK89] N. Immerman and D. Kozen. Definability with bounded number of bound variables. *Information and Computation*, 83(2):121–139, 1989.

[Imm82] N. Immerman. Upper and lower bounds for first order expressibility. *J. Comput. System Sci.*, 25:76–98, 1982.

[Imm91] N. Immerman. $\mathbf{DSPACE}[n^k] = \mathbf{VAR}[k+1]$. Structures '91, 334–340.

[Kam68] J. A. W. Kamp. *Tense Logic and the Theory of Linear Order*. PhD thesis, University of California, Berkeley, 1968.

[PP86] D. Perrin and J.-E. Pin. First-order logic and star-free sets. *J. Comput. System Sci.*, 32:393–406, 1986.

[PW97] J.-E. Pin and P. Weil. Polynomial closure and unambiguous product. *Theory Comput. Systems*, 30:383–422, 1997.

[Sch76] M. P. Schützenberger. Sur le produit de concatenation non ambigu. *Semigroup Forum*, 13:47–75, 1976.

[Str78] H. Straubing. *Varieties of recognizable sets whose syntactic monoids contain solvable groups.* PhD thesis, University of California, Berkely, 1978.

[TW96] D. Thérien and Th. Wilke. Temporal logic and semidirect products: An effective characterization of the until hierarchy. FOCS '96, 256–263.

[Tho82] W. Thomas. Classifying regular events in symbolic logic. *J. Comput. System Sci.*, 25:360–376, 1982.