Regular Languages Defined by Generalized First-Order Formulas with a Bounded Number of Bound Variables

Howard Straubing[1] and Denis Thérien[2]

[1] Computer Science Department, Boston College
Chestnut Hill, Massachusetts, USA 02467
[2] School of Computer Science, McGill University
Montréal, Québec
Canada

Abstract. We consider generalized first-order sentences over $<$ using both ordinary and modular quantifiers. It is known that the languages definable by such sentences are exactly the regular languages whose syntactic monoids contain only solvable groups. We show that any sentence in this logic is equivalent to one using three variables only, and we prove that the languages expressible with two variables are those whose syntactic monoids belong to a particular pseudovariety of finite monoids, namely the wreath product of the pseudovariety **DA** (which corresponds to the languages definable by ordinary first-order two-variable sentences) with the pseudovariety of finite solvable groups. This generalizes earlier work of Thérien and Wilke on the expressive power of two-variable formulas in which only ordinary quantifiers are present. If all modular quantifiers in the sentence are of the same prime modulus, this provides an algorithm to decide if a regular language has such a two-variable definition.

1 Introduction

1.1 Automata and logic

One finds in the theory of finite automata a meeting ground between algebra and logic, where difficult questions about expressibility can be classified, and very often effectively decided, by appeal to the theory of semigroups. This line of research began with the work of McNaughton and Papert [9], who showed that the regular languages definable by first-order sentences over '$<$' are precisely the 'star-free' regular languages, and thus, by a theorem of Schützenberger, the languages whose syntactic monoids are *aperiodic*–that is, contain no nontrivial groups. Let us give an example of the kind of first-order formulas we are talking about. (We will give a more formal account of our approach to logic in the Section 3.) Consider the sentence

$$\exists x \exists y (Q_\sigma x \wedge Q_\sigma y \wedge x < y \wedge \neg \exists z (x < z \wedge z < y)).$$

This formula is interpreted in words over a specified finite alphabet Σ that contains the letter σ. Let us say $\Sigma = \{\sigma, \tau\}$. The variables in the sentence denote positions in the word (that is, integers between 1 and the length of the word, inclusive) and the subformula $Q_\sigma x$ means 'the letter in position x is σ'. The subformula

$$x < y \wedge \neg \exists z (x < z \wedge z < y))$$

says that position x is to the left of position y, and that there is no position strictly between them (*i.e.,* that $y = x + 1$); thus the whole sentence says 'there are two consecutive occurrences of σ'. We say that the sentence defines a language over Σ, namely the set of all strings that contain the factor $\sigma\sigma$. This is the language L given by the regular expression $\Sigma^* \sigma\sigma \Sigma^*$. Since Σ^* is the complement of the empty language, L can be built from the empty language and the letters of Σ by repeated application of boolean operations and concatenation—this is what is meant by a "star-free" language.

The theorem of McNaughton and Papert furnishes an algorithm for determining if a given regular language is definable by such a first-order sentence, since we can compute the multiplication table of the syntactic monoid of a regular language from any automaton that recognizes the language, or from any regular expression that generates it, and we can decide, once we have the multiplication table, whether the monoid contains a nontrivial group.

Since McNaughton and Papert's work, researchers have investigated the expressibility of regular languages in various restrictions and extensions of first-order logic over $<$. For example, we can replace the predicate $x < y$ by the (weaker) predicate $y = x + 1$ (Beauquier and Pin [2]). We can permit the use of *modular* quantifiers, that count, modulo a fixed period, the number of positions of a string satisfying a given condition (Straubing, Thérien and Thomas [24]). We can consider the hierarchy of families of languages parametrized by the depth of quantifier alternation (Thomas [27]). In all the cases considered, the family of regular languages obtained can be characterized in terms of the syntactic monoids or the syntactic morphisms of its members, and in most cases (the quantifier alternation hierarchy is a notable exception) this characterization gives rise to an algebraic algorithm for deciding membership of a given language in the family. The book by Straubing [21] provides a large catalogue of such results.

Kamp [8] and later, Immerman and Kozen [7] showed that every first-order sentence over $<$ is equivalent to such a sentence in which only three variables are used. The number of bound variables that occur in a formula can be considered as a kind of expressibility resource, along with the kinds and depth of the quantifiers and the set of available atomic formulas. (This plays a role in Immerman's treatment [6] of descriptive complexity.)

Thérien and Wilke [26] considered the regular languages defined by sentences in which only two variables are used, and found that these, too, could be characterized in algebraic terms: A language L is definable by a sentence with two variables if and only if its syntactic monoid belongs to a particular family $\mathbf{DA}$ of finite aperiodic monoids. (We will give the precise definition of $\mathbf{DA}$ in the next section.) It was already known that the two-variable definable languages are precisely those definable in the fragment of propositional temporal logic that includes both the past and future versions of the Next and Eventually operators, but excludes the Until operator (Etessami, Vardi and Wilke [4]). Since it is possible to determine from the multiplication table of a finite monoid whether it belongs to **DA**, the Thérien-Wilke result provides an algorithm for determining whether a given regular language is definable in this fragment of temporal logic.

In the present paper we investigate the effect of bounding the number of bound variables in sentences that include modular quantifiers as well as ordinary first-order quantifiers, and we characterize, again in algebraic terms, the regular languages that are thereby defined.

1.2 A critical example

We will establish (Theorem 1 below) that for sentences with both modular and ordinary quantifiers, the three-variable property continues to hold. Thus, by results in [24], the three-variable definable languages are exactly those whose syntactic monoids contain only solvable groups. The real question then is to characterize the languages that are definable by sentences with two variables. It is tempting to conjecture that if a language is definable by a two-variable sentence with modular quantifiers, and if the language is star-free, then it is definable by a two-variable sentence that uses only ordinary quantifiers. But this is false, as the following simple (and important) example shows: Let $\Sigma = \{\sigma, \tau\}$, and let L be the language defined by the regular expression $(\sigma\tau)^*$. Thus $w \in L$ if and only if w contains no occurrence of the factor $\sigma\sigma$ or $\tau\tau$, and, if w is not the empty string, w begins with σ and ends with τ. We already saw in 1.1 how to write a first-order sentence that says a string contains no occurrence of $\sigma\sigma$ or $\tau\tau$. We can say that a string is empty or begins with σ with the sentence:

$$\forall x(\forall y(\neg y < x) \to Q_\sigma x),$$

and we similarly say that a string is empty or ends with τ. Thus L is definable by a first-order sentence. We could have obtained the same conclusion by constructing the syntactic monoid of L and verifying that it contains no nontrivial groups. A closer look a the syntactic monoid shows that the image of the word $\sigma\tau$ under the syntactic morphism is idempotent, but that the image of $\sigma\tau\sigma$ is not idempotent. This implies $M(L) \notin \mathbf{DA}$, so by the theorem of Thérien and Wilke cited above, L cannot be defined by a first-order sentence with only two variables. But L *is* definable by a two-variable sentence if we permit modular quantifiers: The formula $\exists^{r \bmod n} x\phi$ is interpreted to mean 'the number of positions x satisfying ϕ is congruent to r modulo n.' A string belongs to L if and only if it has even length, and has σ in all the odd-numbered positions and τ in all the even-numbered positions. Thus we can define L by the sentence

$$\exists^{0 \bmod 2} x(x = x) \wedge \forall y(Q_\sigma y \leftrightarrow \exists^{0 \bmod 2} x(x < y)).$$

This example shows that the situation is more complicated, and potentially more interesting, than what one might suppose, since the modular quantifiers can be used to economically express properties that are not intrinsically periodic (that is, that do not require modular quantifiers for their expression). This same phenomenon is seen in some strikingly similar results in computational complexity theory, which we will discuss at the end of the paper.

1.3 Our main results

In Section 2, we will give a quick rundown of the terminology and results that we need from semigroup theory. In Section 3, we will give a precise account of our logical formalism, and then prove that the three-variable property holds for formulas that include modular quantifiers as well as ordinary first-order quantifiers. That is,

Theorem 1. *Let ϕ be a sentence over $<$ containing first-order and modular quantifiers. Then ϕ is equivalent to such a sentence with only three variables.*

For formulas that contain *only* modular quantifiers, we have an even stronger result:

Theorem 2. *Let ϕ be a sentence over $<$ in which only modular quantifiers appear. Then ϕ is equivalent to such a sentence with only two variables.*

Our main theorem is that the languages L defined by two-variable sentences are characterized by membership of the syntactic monoid $M(L)$ in the pseudovariety $\mathbf{DA} * \mathbf{G}_{sol}$, defined in Section 2:

Theorem 3. *Let Σ be a finite alphabet. A regular language $L \subseteq \Sigma^*$ is defined by a two-variable sentence over $<$ containing first-order and modular quantifiers, if and only if $M(L) \in \mathbf{DA} * \mathbf{G}_{sol}$*

It is important to remark that while our main theorem permits us in many individual cases to show that a language is, or is not, two-variable definable, the general problem of determining membership in $\mathbf{DA} * \mathbf{G}_{sol}$ is not known to be decidable.

In Section 4, we will prove the 'only if' direction of Theorem 3, by first establishing a normal form for two-variable formulas and then using a version of the Ehrenfeucht-Fraïssé game. In Section 5 we will describe the ideal structure of monoids in $\mathbf{DA} * \mathbf{G}_{sol}$, and then use this information in Section 6 to prove the 'if' part of the theorem. In Section 7 we discuss extensions of the main result, as well as several open problems, most notably the decision problem for $\mathbf{DA} * \mathbf{G}_{sol}$.

2 Background From Semigroup Theory

In this section we give a rapid rundown of the definitions and results from the theory of semigroups that we will need. For further details, the reader should consult the books by Eilenberg [3] and Pin [11].

Division and recognition. A *semigroup* is a set together with an associative multiplication. If the semigroup contains a multiplicative identity element, then we call the semigroup a *monoid,* and we usually denote the identity by 1. If Σ is a finite alphabet, then Σ^*, the set of all strings over Σ, is a monoid with concatenation of strings as the multiplication. Σ^* is the *free monoid* on Σ: This means that if M is any monoid and $f : \Sigma \to M$ any map, then f extends to a unique homomorphism from Σ^* into M. (We note that a homomorphism of monoids is required to preserve the identity as well as the multiplication—that is, the identity element of the domain must map to the identity element of the codomain.)

If S is a finite semigroup and $s \in S$, then there is a unique element e among the powers of s that is idempotent—*i.e.,* such that $e^2 = e$. We denote this idempotent by s^ω.

If M and N are monoids then we say M *divides* N, and write $M \prec N$, if M is a homomorphic image of a submonoid of N. Division is a transitive relation.

If $L \subseteq \Sigma^*$, and M is a monoid, then we say that M *recognizes* L if there is a homomorphism $\phi : \Sigma^* \to M$ and a subset X of M such that $L = \phi^{-1}(X)$. (We also say in this instance that the homomorphism ϕ recognizes L.) L is a regular language if and only if it is recognized a finite monoid. For every $L \subseteq \Sigma^*$, there is a unique monoid $M(L)$ and a homomorphism $\mu_L : \Sigma^* \to M(L)$ such that μ_L recognizes L, and for any homomorphism $\phi : \Sigma^* \to M$ that recognizes L, there is a unique homomorphism $\psi : \phi(\Sigma^*) \to M(L)$ such that $\psi \circ \phi = \mu_L$. (In particular, $M(L)$ divides every monoid that recognizes L, so that L is regular if and only if $M(L)$ is finite.) $M(L)$ is called the *syntactic monoid,* and μ_L the *syntactic morphism,* of L.

Ideal Structure and Green's Relations. If S is a semigroup and I is a nonempty subset of S, then we say I is an *ideal* of S if $SI \subseteq I$ and $IS \subseteq I$. Similarly, we say that I is a *left ideal* of S if $SI \subseteq I$, and a *right ideal* if $IS \subseteq I$. If I is an ideal, then the set $(S - I) \cup \{0\}$ forms a semigroup with multiplication $\times$ given by $s_1 \times s_2 = s_1 s_2$, if $s_1, s_2, s_1 s_2 \in S - I$, and $s_1 \times s_2 = 0$ otherwise. We denote this semigroup by S/I; this is the image of S under the homomorphism that collapses all the elements of I to a single element.

If $s, t \in S$ we write $s \leq_{\mathcal{J}} t$ if s belongs to the ideal $\{t\} \cup St \cup tS \cup StS$ generated by t. If $s \leq_{\mathcal{J}} t$ and $t \leq_{\mathcal{J}} s$, then we say that s and t are *$\mathcal{J}$-equivalent* and write $s \equiv_{\mathcal{J}} t$. The equivalence classes for this relation are called *$\mathcal{J}$-classes.* Similarly we define $\leq_{\mathcal{L}}$, $\leq_{\mathcal{R}}$, $\equiv_{\mathcal{L}}$, $\equiv_{\mathcal{R}}$, *$\mathcal{L}$-class,* and *$\mathcal{R}$-class,* by considering left and right ideals in place of two-sided ideals.

A $\mathcal{J}$-class is said to be *regular* if it contains an idempotent. The *Rees Matrix Theorem,* which we now state, describes the structure of the regular $\mathcal{J}$-classes of a finite semigroup. If A, B are finite sets, G a finite group, and $P : B \times A \to G \cup \{0\}$ a map, then (A, B, G, P) denotes the semigroup $A \times G \times B \cup \{0\}$ with multiplication given by

$$(a, g, b)(a', g', b') = (a, g \cdot P(b, a') \cdot g', b'),$$

if $P(b, a') \neq 0$, and $(a, g, b)(a', g', b') = 0$ otherwise. For each such $\mathcal{J}$-class J of a finite semigroup, there exist finite sets A and B, a finite group G, and a map $P : B \times A \to G \cup \{0\}$ such that the semigroup $J \cup \{0\}$ is isomorphic to (A, B, G, P). Under this isomorphism, the $\mathcal{R}$-classes contained in J are the sets $\{a\} \times G \times B$, the $\mathcal{L}$-classes are the sets $A \times G \times \{b\}$, and every $\mathcal{R}$-class contains at least one idemptotent, as does every $\mathcal{L}$-class. We call (A, B, G, P) a *Rees matrix representation* of J. It is customary to depict J as a matrix whose rows are indexed by A and whose columns are indexed by B.

A non-regular $\mathcal{J}$-class is called a *null* $\mathcal{J}$-class. If J is a null $\mathcal{J}$-class of a finite semigroup and $s, t \in J$, then $st \notin J$.

Wreath Products and Semidirect Products. A *transformation monoid* is a pair $X = (Q, S)$ where X is a set and S is a monoid of maps from Q into itself, with functional composition as the multiplication and the identity map on X as the identity element. We write the image of $q \in Q$ under $s \in S$ as qs, and we compose maps from left to right, so that

$$q(st) = (qs)t$$

for all $q \in Q$, $s, t \in S$. If $Y = (P, T)$ is another transformation monoid, then the *wreath product* $Y \circ X$ is the transformation monoid

$$(P \times Q, T^Q \times S),$$

where we define

$$(p, q)(\alpha, s) = (p \cdot \alpha(q), qs),$$

for all $(p, q) \in P \times Q$, $s \in S$, and $\alpha : Q \to T$. It is straightforward to verify that this set of maps is closed under composition and contains the identity mapping on $P \times Q$.

Let M_1, M_2 be monoids. In order to make the notation more tranparent, we will write the operation in M_1 additively, and denote its identity by 0 (a convention due to Eilenberg [3]). A *left action* of M_2 on M_1 is a map from $M_2 \times M_1$ into M_1, where the image of the pair (m_2, m_1) is denoted $m_2 m_1$, such that

$$m_2(m_1 + m_1') = m_2 m_1 + m_2 m_1',$$

$$(m_2 m_2')m_1 = m_2(m_2' m_1),$$

$$m_2 \cdot 0 = 0,$$

and

$$1 \cdot m_1 = m_1,$$

for all $m_1, m_1' \in M_1$, $m_2, m_2' \in M_2$. We define the *semidirect product* $M_1 * M_2$ relative to this left action as the set $M_1 \times M_2$ together with the multiplication

$$(m_1, m_2)(m_1', m_2') = (m_1 + m_2 m_1', m_2 m_2').$$

It is easy to verify that this multiplication makes $M_1 \times M_2$ into a monoid with identity element $(0, 1)$. The underlying monoid of the wreath product $(P, T) \circ (Q, S)$ is isomorphic to a semidirect product $T' * S$, where T' is a direct product of $|S|$ copies of T.

Pseudovarieties. A *pseudovariety* of finite semigroups is a family of finite semigroups that is closed under finite direct products and division. A pseudovariety of finite monoids is defined analogously. If $\mathbf{V}_1$ and $\mathbf{V}_2$ are pseudovarieties of finite monoids, then $\mathbf{V}_1 * \mathbf{V}_2$ is defined to be the family of all finite monoids M that divide the underlying monoid of a wreath product $(P,T) \circ (Q,S)$, where $T \in \mathbf{V}_1$, $S \in \mathbf{V}_2$. Equivalently, $\mathbf{V}_1 * \mathbf{V}_2$ consists of all divisors of semidirect products $M_1 * M_2$ with $M_1 \in \mathbf{V}_1$, $M_2 \in \mathbf{V}_2$. $\mathbf{V}_1 * \mathbf{V}_2$ is itself a pseudovariety.

If $\mathbf{V}_1$ is a pseudovariety of finite semigroups, and $\mathbf{V}_2$ a pseudovariety of finite monoids, then $\mathbf{V}_1^{-1}\mathbf{V}_2$ consists of all finite monoids M for which there exist finite monoids K, N and homomorphisms $\phi : K \to N$, $\psi : K \to M$, such that ψ maps onto M, $N \in \mathbf{V}_2$, and, for each idempotent $e \in N$, the semigroup $\phi^{-1}(e)$ belongs to $\mathbf{V}_1$. $\mathbf{V}_1^{-1}\mathbf{V}_2$ is a pseudovariety of finite monoids. In this instance we will also say that there is a *relational morphism* $\alpha = \phi \circ \psi^{-1} : M \to N$, and write $\alpha : m \mapsto n$, if there is some $k \in K$ with $\psi(k) = m$, $\phi(k) = n$. Observe that if $\alpha : m_i \mapsto n_i$ for $i = 1, 2$, then $\alpha : m_1m_2 \mapsto n_1n_2$. If $e \in N$ is idempotent, then the set $\alpha^{-1}(e) = \{m \in M : \alpha : m \mapsto e\}$ is a homomorphic image of $\phi^{-1}(e)$, and thus is a semigroup in $\mathbf{V}_1$.

In this paper we will be concerned with the following pseudovarieties of finite monoids:

$\mathbf{A}$–the pseudovariety of finite *aperiodic* monoids; that is, the finite monoids that contain no nontrivial group.
$\mathbf{G}$–the pseudovariety of finite groups.
$\mathbf{G}_{sol}$,–the finite solvable groups.
$\mathbf{R}$–the finite $\mathcal{R}$-trivial monoids; that is, the finite monoids with one-element $\mathcal{R}$-classes.
$\mathbf{J}$–the finite $\mathcal{J}$-trivial monoids; that is, the finite monoids with one-element $\mathcal{J}$-classes.
$\mathbf{J}_1$–the finite commutative monoids in which every element is idempotent.
$\mathbf{DA}$–the finite aperiodic monoids each of whose regular $\mathcal{J}$-classes J is a subsemigroup. (That is, in each Rees matrix representation (A, B, G, P) of J, G is trivial and P never maps to 0.) We will also at times (for example, in the statement of Lemma 4 below) use $\mathbf{DA}$ to denote the pseudovariety of finite semigroups that are subsemigroups of monoids in $\mathbf{DA}$.

Observe that

$$\mathbf{J}_1 \subseteq \mathbf{J} \subseteq \mathbf{R} \subseteq \mathbf{DA} \subseteq \mathbf{A}.$$

We will also consider the pseudovariety $\mathbf{LI}$ of finite semigroups consisting of all semigroups S such that $ese = e$ for all $e, s \in S$ with e idempotent. Such semigroups are called *generalized definite* or *locally trivial* in the literature.

Our main objects of study are pseudovarieties of the form $\mathbf{DA} * \mathbf{H}$, where $\mathbf{H}$ is a pseudovariety of finite groups. There are several alternative characterizations of this pseudovariety:

Lemma 4. *For any pseudovariety* $\mathbf{H}$ *of finite groups,*

$$\mathbf{DA} * \mathbf{H} = \mathbf{DA}^{-1}\mathbf{H} = \mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{H}).$$

Proof. First suppose $M \in \mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{H})$. Then there exist $S \in \mathbf{J}_1$, $H \in \mathbf{H}$, a monoidal semidirect product $S * H$, and a relational morphism $\phi : M \to S * H$ such that the ϕ-preimage of each idempotent in $S * H$ is in $\mathbf{LI}$. Consider the projection homomorphism $\pi : S * H \to H$. The π-preimage of the identity is the set $\{(s, 1) : s \in S\}$. Observe that this set is a submonoid of $S * H$ isomorphic to S; in particular, each element is an idempotent. Thus the $\pi \circ \phi$-preimage of the identity of H is in $\mathbf{LI}^{-1}\mathbf{J}_1$, which is identical to $\mathbf{DA}$ (see Schützenberger[18]). This shows that $\mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{H}) \subseteq \mathbf{DA}^{-1}\mathbf{H}$.

Next, suppose $M \in \mathbf{DA}^{-1}\mathbf{H}$. Then there exist $H \in \mathbf{H}$, and a relational morphism $\psi : M \to H$ such that $\psi^{-1}(1) \in \mathbf{DA}$. We now apply the category-based methods of Tilson [28]: The derived category of the relational morphism ψ has all its base monoids in $\mathbf{DA}$. Since every such locally-$\mathbf{DA}$ finite category is covered by a monoid in $\mathbf{DA}$ (see Almeida [1]) it follows from Derived Category Theorem of [28] that $M \in \mathbf{DA} * \mathbf{H}$.

Finally, suppose that $M \in \mathbf{DA} * \mathbf{H}$. It suffices to prove that every wreath product $M \circ H$, where $M \in \mathbf{DA}$ and $H \in \mathbf{H}$, belongs to $\mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{H})$. Since, as we mentioned above, $\mathbf{DA} = \mathbf{LI}^{-1}\mathbf{J}_1$, there exist a monoid $S \in \mathbf{J}_1$ and a relational morphism $\delta : M \to S$ such that $\delta^{-1}(s) \in \mathbf{LI}$ for all $s \in S$ (since every element of S is idempotent). This induces a relational morphism

$$\delta \circ H : M \circ H \to S \circ H$$

defined as follows:

$$\delta \circ H : (f_1, h_1) \mapsto (f_2, h_2)$$

if and only if $h_1 = h_2$ and $\delta : f_1(h) \mapsto f_2(h)$ for all $h \in H$. The idempotents of $S \circ H$ are the elements of the form $(g, 1)$, where $g : H \to S$, and the $\delta \circ H$-preimage of such an idempotent is

$$\{(f, 1) : \forall h \in H(\delta : f(h) \mapsto g(h))\}.$$

This is simply the direct product of the monoids $\delta^{-1}(g(h))$ over all $h \in H$, and is thus in $\mathbf{LI}$. So $M \circ H \in \mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{H})$, as required.

We will also need some facts about the pseudovarieties $\mathbf{R}*\mathbf{G}$ and $\mathbf{J}_1*\mathbf{G}$: $\mathbf{R}*\mathbf{G}$ consists of all finite monoids such that in each regular $\mathcal{J}$-class there is exactly one idempotent in each $\mathcal{R}$-class. If $\mathbf{H}$ is a pseudovariety contained in $\mathbf{G}$ and closed under semidirect product (for example, $\mathbf{H} = \mathbf{G}_{sol}$) then $\mathbf{H} * \mathbf{R} * \mathbf{H} = \mathbf{R} * \mathbf{H}$. (For the proofs of these last two facts, see Stiffler [20].)

For every finite semigroup S there is a *reversed* semigroup S^{rev}, with the same underlying set as S, and with multiplication $\times$ given by

$$s \times t = ts$$

for all $s, t \in S$. If $\mathbf{V}$ is a pseudovariety of semigroups or monoids, then $\mathbf{V}^{rev}$ denotes the pseudovariety consisting of the reversals of members of $\mathbf{V}$. Every group is isomorphic to its own reversal, via the anti-isomorphism $g \mapsto g^{-1}$. Rhodes and Tilson [15] show that if $\mathbf{H}$ is a pseudovariety contained in $\mathbf{G}$, then for any pseudovariety $\mathbf{V}$, $\mathbf{V} * \mathbf{H} = \mathbf{V} \Box \mathbf{H}$, where $\Box$ denotes a symmetric version of the wreath product. It follows readily that $(\mathbf{R} * \mathbf{H})^{rev} = \mathbf{R}^{rev} * \mathbf{H}$, and so

$$\mathbf{J}_1 * \mathbf{H} \subseteq (\mathbf{R} * \mathbf{H})^{rev} \cap \mathbf{R} * \mathbf{H}.$$

Thus if $M \in \mathbf{J}_1 * \mathbf{H}$, and J is a regular $\mathcal{J}$-class of M, then there is exactly one element in each $\mathcal{L}$-class and in each $\mathcal{R}$-class of M. We also have

$$(\mathbf{J}_1 * \mathbf{H})^{rev} = \mathbf{J}_1^{rev} * \mathbf{H} = \mathbf{J}_1 * \mathbf{H},$$

so that $\mathbf{J}_1 * \mathbf{H}$ is closed under reversal. The same argument shows $\mathbf{DA} * \mathbf{H}$ is closed under reversal as well.

3 Defining Formal Languages in Generalized First-Order Logic

3.1 Word structures

Here we give a brief account of our particular approach to model-theoretic notions. Our development follows Straubing [21]. Let Σ be a finite alphabet. Let $\mathcal{V}$ be a finite subset of the set of variables $\{x_1, x_2, \ldots, \}$. A *word structure* over $(\Sigma, \mathcal{V})$ is a pair (w, I), where $w \in \Sigma^*$ and I is a map from $\mathcal{V}$ into $\{1, \ldots, m\}$, where $m = |w|$. We allow $\mathcal{V} = \emptyset$—in which case we identify the word structure with the word w—and $m = 0$, in which case $\mathcal{V}$ must be the empty set and w the empty word.

We build formulas from the unary predicate symbols $\{Q_\sigma : \sigma \in \Sigma\}$, the binary predicate symbol $<$, the variable symbols $x_1, x_2, \ldots$, the boolean connectives $\neg$ and $\wedge$, and two kinds of quantifier symbols; $\exists$ and $\exists^{r \bmod m}$, where $0 \leq r < m$. The atomic formulas are those of the form $x_j < x_k$ and $Q_\sigma x_j$, where $\sigma \in \Sigma$.

We will suppose at the outset that we never re-use a variable symbol within a formula. That is, the same variable symbol x_j cannot occur in the formula bound by two different quantifiers, or have both a bound and a free occurrence. Now, since this paper is all about what happens when we *do* re-use variable symbols, we will soon drop this convention, but we need it for now to define the semantics of our formulas. We will interpret a formula ϕ in a word structure (w, I) over $(\Sigma, \mathcal{V})$, where the set of free variable symbols in ϕ is contained in $\mathcal{V}$, and the set of bound variable symbols is disjoint from $\mathcal{V}$. Let

$$w = \sigma_1 \cdots \sigma_m \in \Sigma^*.$$

We write

$$(w, I) \models Q_\sigma x_j$$

and say (w, I) *satisfies* $Q_\sigma x_j$, if

$$\sigma_{I(x_j)} = \sigma.$$

We say

$$(w, I) \models x_j < x_k$$

if $I(x_j) < I(x_k)$,

$$(w, I) \models \phi \wedge \psi$$

if (w, I) satisfies both ϕ and ψ,

$$(w, I) \models \neg\phi$$

if (w, I) does not satisfy ϕ,

$$(w, I) \models \exists x_j \phi$$

if

$$(w, \hat{I}) \models \phi,$$

for some extension of $\hat{I}$ of I to a map from $\mathcal{V} \cup \{x_j\}$ into $\{1, \ldots, m\}$, and

$$(w, I) \models \exists^{r \bmod n} x_j \phi$$

if the number of such extensions $\hat{I}$ is congruent to r modulo n. (Informally, this formula says 'the number of positions x_j for which ϕ holds is congruent to r modulo n.')

In our subsequent discussion, we will use the boolean connectives $\vee, \rightarrow, \leftrightarrow$ as well—these can all be defined in terms of $\wedge$ and $\neg$. We will also use the universal quantifier symbol $\forall$—this is definable in terms of $\exists$.

A *sentence* is a formula without free variables. If ϕ is a sentence, then the set

$$L_\phi = \{w \in \Sigma^* : w = (w, \emptyset) \models \phi\}$$

is called the *language defined by* ϕ.

Example. Let L be the set of strings over $\Sigma = \{\sigma, \tau\}$ that contain an even number of factors of the form $\sigma\sigma$. L is defined by the sentence

$$\exists^{0 \bmod 2} x_1 \exists x_2 (x_2 = x_1 + 1 \wedge Q_\sigma x_1 \wedge Q_\sigma x_2\},$$

where '$x_2 = x_1 + 1'$ is an abbreviation for

$$x_1 < x_2 \wedge \neg\exists x_3((x_1 < x_3) \wedge (x_3 < x_2)).$$

We will define a number of operations on formulas, which we call *relativizations.* Let ϕ be a formula in which the variable x_k does not appear. The formula $\phi[< x_k]$ is constructed recursively by beginning with the outermost quantifiers of ϕ, and replacing each subformula

$$\exists^* x_j \psi,$$

where $\exists^*$ is either an ordinary existential quantifier or a modular quantifier, by

$$\exists^* x_j (x_j < x_k \wedge \psi[< x_k]).$$

(If ψ is an atomic formula then $\psi[< x_k]$ is identical to ψ.) Informally, $\phi[< x_k]$ means 'the prefix consisting of the positions to the left of x_k satisfies ϕ'. More precisely, let $w \in \Sigma^*$ and let v be a proper prefix of w. Let $\mathcal{V}$ be a set of variables that does not contain x_k, and let $I : \mathcal{V} \to \{1, \ldots, |v|\}$ be a map. Let $\hat{I}$ be the extension of I to $\mathcal{V} \cup \{x_k\}$ defined by setting $\hat{I}(x_k) = |v| + 1$. Then $(v, I) \models \phi$ if and only if $(w, \hat{I}) \models \phi[< x_k]$.

We define relativizations $\phi[\leq x_k]$, $\phi[> x_k]$, and $\phi[\geq x_k]$ analogously.

Let θ be a *sentence* with the property that $w \models \theta$ if and only if every prefix v of w satisfies θ. Let ϕ be a formula. We define the relativized formula $\phi[\leq \theta]$ by recursively replacing each subformula $\exists^* x_j \psi$ by

$$\exists^* x_j(\theta[\leq x_j] \wedge \psi[\leq \theta]).$$

Let $w \in \Sigma^*$, and let v be the longest prefix of w that satisfies θ. Let $\mathcal{V}$ be a set of variables and let I be a map from $\mathcal{V}$ into $\{1, \ldots, |v|\}$. Then $(v, I) \models \phi$ if and only if $(w, I) \models \phi[\leq \theta]$. In particular, if ϕ is a sentence, then v satisfies ϕ if and only if w satisfies $\phi[< \theta]$.

Let ϕ, θ be as in the preceding paragraph. We define the relativized formula $\phi[> \theta]$ by recursively replacing each quantified subformula $\exists^* x_j \psi$ by

$$\exists^* x_j(\neg\theta[\leq x_j] \wedge \psi[> \theta]).$$

Let $w \in \Sigma^*$, and let $w = vv'$, where v is the longest prefix of w that satisfies θ. Let $\mathcal{V}$ be a set of variables, and let $I : \mathcal{V} \to \{1, \ldots, |v'|\}$ be a map. Let $I' : \mathcal{V} \to \{1, \ldots, |w|\}$ be the map defined by $I'(x_j) = I'(x_j) + |v|$ for all $x_j \in \mathcal{V}$. Then $(v', I) \models \phi$ if and only if $(w, I') \models \phi[> \theta]$. In particular, if ϕ is a sentence, then $v' \models \phi$ if and only if $w \models \phi[> \theta]$.

Example. Let $\Sigma = \{\sigma, \tau\}$, and let L be the set of strings over Σ^* given by the regular expression

$$((\sigma\tau)^* + \tau(\sigma\tau)^*)\sigma\sigma\Sigma^*.$$

Let L' be the set of strings that contain no occurrence of either $\sigma\sigma$ or $\tau\tau$. L' is defined by the following sentence θ :

$$\neg\exists x_1 \exists x_2(x_2 = x_1 + 1 \wedge (Q_\sigma x_1 \leftrightarrow Q_\sigma x_2)).$$

L' has the property that $w \in L'$ if and only if every prefix of w is in L'. The following sentence ϕ says 'the first letter is σ':

$$\exists x_3(Q_\sigma x_3 \wedge \neg\exists x_4(x_4 < x_3)).$$

It follows that L is defined by the sentence

$$\phi[> \theta].$$

According to the foregoing discussion, this sentence is

$$\exists x_3(\neg\theta[\leq x_3] \wedge Q_\sigma x_3 \wedge \neg\exists x_4(\neg\theta[\leq x_4] \wedge x_4 < x_3)),$$

where $\theta[\leq x_3]$ is given by

$$\neg\exists x_1(x_1 \leq x_3 \wedge \exists x_2(x_2 \leq x_3 \wedge x_2 = x_1 + 1 \wedge (Q_\sigma x_1 \leftrightarrow Q_\sigma x_2))).$$

Observe that the subformula $x_1 \leq x_3$ of the above formula can be eliminated.

3.2 Three-variable formulas and the proof of Theorem 1

Consider the following sentence:

$$\exists x \exists y (x < y \wedge \exists x (y < x \wedge \exists y (x < y))).$$

Strictly speaking, the meaning of this sentence is not defined in our formalism, since it violates our convention about the re-use of variable symbols. Nonetheless, the meaning of the sentence is clear. We can write an equivalent sentence that is consistent with our formalism by choosing a set of variable symbols in one-to-one correspondence with the quantifier symbols of the sentence, and replacing each variable symbol v of the sentence by the new variable symbol x_i corresponding to the quantifier that binds v. This gives the sentence:

$$\exists x_1 \exists x_2 (x_1 < x_2 \wedge \exists x_3 (x_2 < x_3 \wedge \exists x_4 (x_3 < x_4))).$$

Thus the the sentence says that the string contains four distinct positions; i.e., it defines the set of strings over Σ of length at least 4. Thus, when we permit re-use of variable symbols, this language is definable with two variables.

Kamp [8], and later Immerman and Kozen [7], showed that every sentence over our base of atomic formulas that uses only ordinary (as opposed to modular) quantifiers is equivalent to a sentence with at most three variables. Here we prove Theorem 1, stated in 1.3: that the reduction to three variables is possible for sentences with modular quantifiers as well. To this end we consider sequences

$$x_{\pi(1)} < \cdots < x_{\pi(k)},$$

where π is a permutation of $\{1, \ldots, k\}$. Such a sequence is said to be a *good* sequence if it is built in the following fashion: We set α_1 to be the sequence

$$x_k,$$

and, for $1 \leq i < k$, we form α_{i+1} either by adjoining '$x_{k-i+1} <$' to the left of α_i, or '$< x_{k-i+1}$' to the right of α_i. α_k is then a good sequence.
Example. With $k = 6$ we build a good sequence as follows:

$$x_6$$

$$x_5 < x_6$$

$$x_4 < x_5 < x_6$$

$$x_4 < x_5 < x_6 < x_3$$

$$x_4 < x_5 < x_6 < x_3 < x_2$$

$$x_1 < x_4 < x_5 < x_6 < x_3 < x_2.$$

A *three-variable definition* of a sequence is constructed as follows: The first line is always

$$x.$$

We say that x is *bound to* x_1. The second line is either

$$x < y$$

or

$$y < x,$$

and we say that y is bound to x_2. The third line is one of the three permutations of

$$x < y < z$$

that is consistent with the second line. We say that z is bound to x_3. For $j > 3$, the j^{th} line has one of the following three forms:
(a)

$$u : v < u < w,$$

where u, v, w are the three variables x, y, z in some order, and $v < w$ appears within the preceding line. We say that u is *unbound* from whatever variable x_i it was previously bound to, and is now bound to x_j.
(b)

$$u : u < v,$$

where u and v are distinct variables, and v is bound to the least element of $x_1, \ldots, x_{j-1}$. We say that u is unbound from whatever variable it was previously bound to, and is now bound to x_j.
(c)

$$u : v < u,$$

where u and v are distinct variables, and v is bound to the greatest element of $x_1, \ldots, x_{j-1}$. We say that u is unbound from whatever variable it was previously bound to, and is now bound to x_j.

Such a three-variable definition with k lines defines a unique ordering on $x_1, \ldots, x_k$.

Example. Consider the following three-variable definition:

$$x$$
$$x < y$$
$$x < z < y$$
$$y : x < y < z$$
$$x : y < x < z$$
$$y : x < y < z$$

This defines the sequence

$$x_1 < x_4 < x_5 < x_6 < x_3 < x_2$$

of the previous example.

Lemma 5. *Every good sequence is defined by a three-variable definition.*

Proof. We prove this by induction on the number of variables in the good sequence. First note that the claim is obviously true if the good sequence involves only the variables x_1, x_2, x_3. Second, the subsequence of a good sequence on $x_1, \ldots, x_k$ that we obtain by eliminating x_k is itself a good sequence on $x_1, \ldots, x_{k-1}$. Suppose that we have a three-variable definition D for this smaller sequence; it suffices to show that we can extend the definition to the larger sequence. If x_k is the least element of the sequence, then by the rules of good sequence formation, the entire sequence must be

$$x_k < x_{k-1} < \cdots < x_1.$$

In particular, x_{k-1}, introduced on the last line of D, is bound to one of the three variables x, y, z. Let us say it is bound to x. Then we can extend the definition by adjoining the line:

$$y : y < x.$$

The analogous reasoning applies to the case where x_k is the largest element. So suppose now that x_k appears between two elements of the sequence; that is

$$x_i < x_k < x_j,$$

and that there are no elements of the sequence between x_i and x_k, or between x_k and x_j. One of i and j must be $k-1$, for if not, we would have a situation like

$$x_{k-1} < \cdots < x_i < x_k < x_j,$$

which violates the rules of good sequence formation. Assuming $i = k-1$, then, our sequence must have the form

$$x_{k-(r-1)} < \cdots < x_{k-1} < x_k < x_j < \cdots,$$

with $j = k-r$. (We get a similar conclusion if we assume instead that $j = k-1$.) This implies that one of the variables x, y, z is bound to x_j, and another of the variables is bound to x_{k-1}. Let's say x is bound to x_j, and y to x_{k-1}. So we can extend D by adding the line:

$$z : y < z < x.$$

Thus in all cases we have a three-variable definition of the larger sequence.

We now apply the preceding lemma to prove Theorem 1. Every language in Σ^* defined by one of our sentences is a regular language whose syntactic monoid contains only solvable groups. (Straubing, Thérien and Thomas [24].) Further, every such language can be built by beginning with the languages

$$\Sigma^* \sigma \Sigma^*,$$

where $\sigma \in \Sigma$, and applying boolean operations and the operations

$$(L_1, L_2) \mapsto L_1 \sigma L_2,$$

where $\sigma \in \Sigma$, and

$$(L_1, L_2) \mapsto (L_1, L_2, \sigma, r, m),$$

where $\sigma \in \Sigma$, $0 \leq r < m$, and the right-hand side denotes the set of strings w for which the number of factorizations $w = w_1 \sigma w_2$ with $w_1 \in L_1$ and $w_2 \in L_2$ is congruent to r modulo m. (See, for example, Straubing [23] or Thérien [25].) Thus if $L \subseteq \Sigma^*$ is definable by a sentence, we can construct an equivalent sentence as follows: We begin with the sentences

$$\exists x_n Q_\sigma x_n,$$

which define the languages $\Sigma^* \sigma \Sigma^*$. Suppose now that we have sentences ϕ_1 and ϕ_2, with disjoint sets of variables in $\{x_{k+1}, \ldots, x_n\}$ defining languages L_1 and L_2, respectively. Then $L_1 \cap L_2$ is defined by $\phi_1 \wedge \phi_2$, $\Sigma^* - L_1$ is defined by $\neg\phi_1$, $L_1 \sigma L_2$ is defined by

$$\exists x_k (Q_\sigma x_k \wedge \phi_1[< x_k] \wedge \phi_2[> x_k]),$$

and (L_1, L_2, σ, r, n) is defined by

$$\exists^{r \bmod n} x_k (Q_\sigma x_k \wedge \phi_1[< x_k] \wedge \phi_2[> x_k]).$$

It follows that L is defined by a sentence in which each quantified subformula has the form

$$\exists^* x_k (s(x_1, \ldots, x_k) \wedge Q_\sigma x_k \wedge \alpha),$$

where $s(x_1, \ldots, x_k)$ is a good sequence on some subset of $\{x_1, \ldots, x_k\}$, and α is a boolean combination of quantified formulas $\exists^* x_{k+1} \psi$ of the same form, whose good sequences extend s. The reason we obtain good sequences is that whenever we add a new variable symbol in constructing the relativized formulas, we adjoin the symbol either to the beginning or to the end of the existing sequences. We can now apply Lemma 5 and rewrite the sentence, beginning with the outermost quantifers, as an equivalent three-variable sentence.

3.3 The two-variable property for $\mathbf{G}_{sol}$ and $\mathbf{R} * \mathbf{G}_{sol}$.

Our main theorem implies that every regular language whose syntactic monoid is in $\mathbf{DA} * \mathbf{G}_{sol}$ is definable by a sentence that uses only two variables. Here we prove this theorem in some easy special cases, which we will use in Section 6 in the proof of the general result.

First we consider the languages in Σ^* defined by sentences in which *only* modular quantifiers are used. These are precisely the regular languages whose syntactic monoids belong to $\mathbf{G}_{sol}$. In [23] it is shown that this class of languages is the closure of the empty language under boolean operations and the operations

$$L \mapsto (L, \Sigma^*, \sigma, r, n),$$

where $\sigma \in \Sigma$ and $0 \leq r < n$. If ϕ is a sentence that defines the language L, then $(L, \Sigma^*, \sigma, r, n)$ is defined by the sentence

$$\exists^{r \bmod n} x(Q_\sigma x \wedge \phi[< x]).$$

The empty language is defined by the sentence

$$\exists^{1 \bmod 2} x(Q_a x \wedge \neg Q_a x).$$

It follows that if $M(L) \in \mathbf{G}_{sol}$, then we can construct a sentence defining L by relativizing exclusively on the left. This means (see the discussion in 3.2) that the good sequences that arise are all of the form

$$x_n < \cdots < x_2 < x_1.$$

Plainly, such a sequence admits a two-variable definition

$$\begin{gathered} x \\ y < x \\ x : x < y \\ y : y < x \\ \cdot \\ \cdot \\ \cdot \end{gathered}$$

so we can proceed as in 3.2, and obtain a two-variable sentence for L. This proves Theorem 2, that the languages defined by sentences that use modular quantifiers alone are defined by such sentences in which only two variables appear.

The very same argument shows that every member of the smallest family of languages closed under boolean operations and the operations

$$L \mapsto (L, \Sigma^*, \sigma, r, n)$$

and

$$L \mapsto L\sigma\Sigma^*$$

is definable by a two-variable sentence, using both modular and ordinary quantifiers. It follows from results of Stiffler [20] that this is precisely the family of languages whose syntactic monoids belong to the pseudovariety $\mathbf{R} * \mathbf{G}_{sol}$. Observe that the sentence ϕ that we constructed has the additional property that the formula $\phi[< x]$, which has one free variable, is equivalent to a formula with only two variables. We say in this instance that ϕ is a *left-relativizable* two-variable sentence. We state the results of this subsection formally for future reference:

Theorem 6. *If $L \subseteq \Sigma^*$ is a regular language with $M(L) \in \mathbf{R} * \mathbf{G}_{sol}$, then L is definable by a left-relativizable two-variable sentence.*

4 The Syntactic Monoid of Two-variable Definable Languages

In this section we prove that every regular language definable by a two-variable sentence has its syntactic monoid in $\mathbf{DA} * \mathbf{G}_{sol}$. This is one direction of our main result, Theorem 3.

4.1 A normal form for two-variable formulas

The first step in our proof is to show that if $\theta(x)$ is a two-variable formula with x free, then θ is equivalent to a two-variable formula in which *an ordinary quantifier never appears within the scope of a modular quantifier.* It is sufficient to show that if $\alpha(x,y)$ is a two-variable formula with x, y free, in which an ordinary quantifier does not appear within the scope of a modular quantifier, then

$$\exists^{r \bmod n} y \alpha(x,y)$$

is equivalent to a formula in this normal form.

Since α is a boolean combination of atomic formulas and quantified formulas in which the bound variable is either x or y, we can write it as a disjunction of three formulas of the form

$$x\mathcal{R}y \wedge \phi(x) \wedge \psi(y),$$

where $\mathcal{R}$ ranges over the relations $<$,$>$ and $=$, y is not free in ϕ and x is not free in ψ. Observe that if $\delta_1, \delta_2, \delta_3$ are disjoint formulas (that is, never satisfied by the same word structure) then

$$\exists^{r \bmod n} y(\delta_1 \vee \delta_2 \vee \delta_3)$$

is equivalent to a boolean combination of formulas $\exists^{s \bmod n} y \delta_i$. Thus we wind up with a boolean combination of formulas

$$\exists^{s \bmod n} y(x\mathcal{R}y \wedge \phi(x) \wedge \psi(y)).$$

Since y is not free in ϕ, the above formula is equivalent to

$$\phi(x) \wedge \exists^{s \bmod n} y(x\mathcal{R}y \wedge \psi(y)),$$

if $s \neq 0$. If $s = 0$ then it is equivalent to

$$\neg\phi(x) \vee \neg\exists y(x\mathcal{R}y \wedge \psi(y)) \vee \exists^{0 \bmod n} y(x\mathcal{R}y \wedge \psi(y)).$$

The formula

$$\exists^{s \bmod n} y(x = y \wedge \psi(y))$$

is equivalent to $\psi(x)$ if $s = 1$, and is never satisfied otherwise, so we may now restrict our attention to formulas of the form

$$\exists^{s \bmod n} y(x\mathcal{R}y \wedge \psi(y)),$$

where $\mathcal{R}$ is either $<$ or $>$. We will assume that $\mathcal{R}$ is $<$; the other case is treated similarly.

The formula $\psi(y)$ is itself a boolean combination of $Q_\sigma y$ and quantified formulas. By taking boolean combinations again, we can reduce to formulas of the form

$$\gamma(x) : \exists^{t \bmod n} y(x < y \wedge Q_\sigma y \wedge \beta_1(y) \wedge \cdots \wedge \beta_k(y)),$$

where each β_i is a quantified formula (with either $\exists$, $\forall$, or a modular quantifier). We will show how to eliminate the β_i that begin with ordinary quantifiers, so that in the end we are left with only those β_i that begin with modular quantifiers. By the inductive hypothesis, these contain no ordinary quantifiers within the scope of modular quantifiers.

First we introduce a new notation. A word structure (w, I) for which the domain of I contains a single variable v will be denoted (w, i), where $i = I(v)$. We will consider several possible forms for β_1. First,

$$\beta_1(y) : \exists x(x < y \wedge \psi(x)).$$

If (w, i) is a word structure, then one of the following three conditions must hold:
(i)There is no position j such that $(w, j) \models \psi(x)$.
(ii)There is such a position, and the least such position is less than or equal to i.
(iii) There is such a position, and the least such position is greater than i.

In case (i), $(w, i) \models \gamma(x)$ if and only if $t = 0$ and

$$(w, i) \models \neg\exists y \psi(y).$$

Note that the modular quantifier disappears. In case (ii), $(w, i) \models \gamma(x)$ if and only if

$$(w, i) \models \exists y(y \leq x \wedge \psi(y)) \wedge \\ \exists^{t \bmod n} y(x < y \wedge Q_\sigma y \wedge \bigwedge_{s=2}^{n} \beta_s(y)).$$

In case (iii), $(w, i) \models \gamma(x)$ if and only if

$$(w, i) \models \exists y(\psi(y) \wedge \forall x(x < y \to \neg\psi(y)) \wedge \\ x < y \wedge \exists^{t \bmod n} x(y < x \wedge Q_\sigma x \wedge \bigwedge_{s=2}^{n} \beta_s(x)).$$

$\gamma(x)$ is thus equivalent to the disjunction of these three formulas. Note that in all cases we have reduced the number of β_s within the scope of modular quantifiers.

Suppose now that $\beta_1(y)$ has the form

$$\exists x(x > y \wedge \psi(x)).$$

We consider two possibilities for a word structure (w, i):
(i) There is no $j > i$ such that $(w, j) \models \psi(x)$.

(ii) There exists $j > i$ such that $(w,j) \models \psi(x)$.

In the first case, $(w,i) \models \gamma(x)$ if and only if $t = 0$ and

$$(w,i) \models \forall y(y > x \rightarrow \neg\psi(x)).$$

In the second case, consider the greatest j such that $(w,j) \models \psi(x)$. Now for (w,i) to satisfy γ, the number of positions *between* i and j that contain a σ and satisfy the remaining β_s must be congruent to t modulo n. This would appear to require the introduction of a third variable, but now we get to use the fact that $\mathbf{Z}_n$ is a group: We find that $(w,i) \models \gamma(x)$ if and only if (w,i) satifies the disjunction, over all pairs (t_1, t_2) such that $t_1 - t_2 = t$ in $\mathbf{Z}_n$, of the formulas

$$\exists y(y > x \wedge \psi(y) \wedge \forall x(x > y \rightarrow \neg\psi(y)) \wedge \exists^{t_2 \bmod n} x(x \geq y \wedge Q_\sigma x \wedge \bigwedge_{s=2}^{k} \beta_s(x)))$$
$$\wedge\ \exists^{t_1 \bmod n} y(y > x \wedge Q_\sigma y \bigwedge_{s=2}^{k} \beta_s(y))$$

Note that once again, we reduce the number of β_s within the scope of the modular quantifier.

The other two possible forms for β_1 are

$$\forall x(x < y \rightarrow \psi(x))$$

and

$$\forall x(x > y \rightarrow \psi(x)).$$

These are handled similarly to the two preceding cases. In all instances, we reduce the number of β_s within the scope of the modular quantifier. We continue in this way with β_2, β_3, *etc.* until the only β_s remaining within the scope of the modular quantifiers are themselves formulas that use only modular quantifiers.

Example. Let us illustrate the foregoing argument with an example. We consider the language L over the alphabet $\{a, b\}$ such that the number of b's that have at least two a's to the left of it and one a to the right of it is odd. L is defined by the sentence

$$\exists^{0 \bmod 2} x(Q_b x \wedge \beta_1(x) \wedge \beta_2(x)),$$

where $\beta_1(x)$, which says that there are at least two a's to the left of x, is

$$\exists y(Q_a y \wedge y < x \wedge \exists x(x < y \wedge Q_a x)),$$

and $\beta_2(x)$, which says that there is at least one a to the right of x, is

$$\exists y(Q_a y \wedge y > x).$$

Observe that our sentence has two variables. The new sentence will say, in effect, that there are at least three a's (which we can express with two variables, using only ordinary quantifiers) and that either the number of b's following the second a is odd and the number of b's following the last a is even, or the number of b's following the second a is even and the number of b's following the last a is odd. Each of these conditions can be expressed as a two-variable sentence in which the modular quantifiers are pushed to the bottom level. For example, to say that the number of b's following the second a is even we use the sentence

$$\exists x(\alpha_1(x) \wedge \alpha_2(x) \wedge \exists^{0 \bmod 2} y(y > x)).$$

The formula $\alpha_1(x)$ is

$$Q_a x \wedge \exists y(y < x \wedge Q_a y)$$

and $\alpha_2(x)$ is

$$\neg \exists y(y < x \wedge Q_a y \wedge \exists x(x < y \wedge Q_a x)).$$

4.2 Games and Formulas

Let us fix a modulus m and a depth r, and let us treat as atomic formulas all two-variable formulas with one free variable using exclusively modular quantifiers of modulus m and depth no more than r. Observe that there are only finitely many inequivalent formulas of this form.

We look at two-variable first-order formulas over this base of atoms. By the *depth* of such a formula we mean the depth of nesting of the ordinary first-order quantifiers. Because of our normal form result in the preceding subsection, it is sufficient to prove that the syntactic monoid of any language defined by such a formula is in $\mathbf{DA} * \mathbf{G}_{sol}$.

For each $k \geq 0$ we define two equivalence relations, one on words, and one on word structures of the form (w, i), both denoted $\equiv_k$:

$$w_1 \equiv_k w_2$$

if and only if w_1 and w_2 satisfy the same sentences of depth k or less. For word structures,

$$(w_1, i) \equiv_k (w_2, j)$$

if and only if the two structures satisfy the same formulas $\phi(x)$ (with one free variable) of depth k or less.

Let us give an explict description of $\equiv_0$: Let $\mathbf{H}$ be the pseudovariety of finite abelian groups of exponent m, and let $\mathbf{H}^r$ be the pseudovariety consisting of all finite groups that have a normal series of length r or less in which every quotient group belongs to $\mathbf{H}$. For every finite alphabet Σ, $\mathbf{H}^r$ has a finite Σ-generated free object F, and there is the canonical homomorphism $\pi : A^* \to F$. It follows from results in [21] that two words are $\equiv_0$-equivalent if and only if they have the same image under π. Furthermore, two structures (w_1, i) and (w_2, j) are $\equiv_0$-equivalent if and only if there are factorizations

$$w_1 = u\sigma v$$

and

$$w_2 = u'\sigma v'$$

where $\sigma \in \Sigma$, $|u| = i-1$, $|u'| = j-1$, $\pi(u) = \pi(u')$ and $\pi(v) = \pi(v')$. From this follows the important fact that not only is $\equiv_0$ a congruence on words, but it is a *congruence on structures* in the sense that if

$$(w_1, i) \equiv_0 (w_2, j),$$

$$u_1 \equiv_0 u_2,$$

and

$$v_1 \equiv_0 v_2,$$

then

$$u_1(w_1, i)v_1 \equiv_0 u_2(w_2, j)v_2.$$

($u_1(w_1, i)v_1$ is shorthand for $(u_1 w v_1, i + |u_1|)$.)

For $k > 0$, we characterize $\equiv_k$ in terms of a version of the Ehrenfeucht-Fraïssé game introduced by Wilke [30]. The game is played on two structures (w_1, i) and (w_2, j). If these are not $\equiv_0$-equivalent, then Player I wins at once, in zero rounds. Otherwise, each round proceeds as follows. Think of each structure as a word with a pebble on one position. Player I picks one of the words and moves the pebble one or more positions to the left or right. For example, he might pick (w_1, i) and change it to (w_1, i'), where $i < i'$. Player II must now move the pebble in the other string in the same direction (left if Player I moved left, right if Player I moved right). In this example, Player II must produce (w_2, j') with $j < j'$. The two new structures are required to be $\equiv_0$-equivalent—Player II loses if she cannot meet this requirement. If Player II can correctly respond for k successive rounds, then she wins the game.

We can also play the game on words. In the first round, Player I places his pebble on a position in one of the words, and Player II pebbles a position in the other word. The resulting structures (w_1, i) (w_2, j) are required to be $\equiv_0$-equivalent, or Player II loses. Play then proceeds as above for $k-1$ additional rounds.

We now show that the standard result for model-theoretic games holds for this variant.

Lemma 7. *$(w_1, i) \equiv_k (w_2, j)$ if and only if Player II has a winning strategy in the k-round game on these two structures. $w_1 \equiv_k w_2$ if and only if Player II has a winning strategy in the k-round game on these two words.*

Proof. First we prove by induction that if $k > 0$ and $(w_1, i) \equiv_k (w_2, j)$, then Player II has a winning strategy in the k-round game on these two structures. Note that the base case $k = 0$ is trivial. Suppose Player I makes his first move in (w_1, i) and moves the pebble to the right. (The three other cases are treated analogously.) The result is a structure (w_1, i'), where $i < i'$. The $\equiv_{k-1}$-class of (w_1, i') is defined by a two-variable formula $\psi(x)$ of depth $k-1$ having one free variable. We then have $(w_1, i) \models \exists y(x < y \wedge \psi(y))$, which is a two-variable formula. By assumption (w_2, j) satisfies the same formula, so there exists $j' > j$

such that $(w_1,i') \equiv_{k-1} (w_2,j')$. The inductive hypothesis implies that Player II has a winning strategy in the $(k-1)$-round game on these two structures. Thus Player II replies to Player I's first move by moving the pebble on w_2 to position j', and thereafter plays her winning strategy in the $(k-1)$-round game.

Next we prove by induction that if (w_1,i) and (w_2,j) are *not* $\equiv_k$-equivalent, then Player I has a winning strategy in the k-round game. Once again, the case $k=0$ is trivial. The non-equivalence implies that there is a two-variable formula

$$\exists x(x\mathcal{R}y \wedge \phi(x) \wedge \psi(y))$$

where $\mathcal{R}$ denotes $<$ or $>$, such that ϕ and ψ have depth less than k, x is not free in ψ and y is not free in ϕ, and such that one of the structures satisfies the formula and the other does not. We can suppose without loss of generality that (w_1,i) satisfies the formula and that $\mathcal{R}$ is $<$. We can pull $\psi(y)$ outside the quantifier. Observe that if (w_2,j) does not satisfy $\psi(y)$, then by the inductive hypothesis, Player I has a winning strategy in the $(k-1)$-round game in the two structures, and hence in the k-round game. Thus we can suppose that the formula is

$$\exists x(x<y \wedge \phi(x)).$$

Player I begins by moving the pebble in w_1 to a position $i' < i$ such that $(w_1,i') \models \phi(x)$. We are supposing that there is no position $j' < j$ such that $(w_2,j') \models \phi(x)$, so wherever Player II moves, the resulting pointed word (w_2,j') will not be $(k-1)$-equivalent to (w_1,i'). Thus, by the inductive hypothesis, Player I has a strategy that will within the next $k-1$ rounds.

Now let us consider the game for ordinary words. If $w_1 \equiv_k w_2$ and Player I makes his initial placement in w_1, then the resulting structure (w_1,i) satisfies $\phi(x)$, where $\phi(x)$ is a two-variable formula of depth $k-1$ that defines the $\equiv_{k-1}$-class of (w_1,i). Thus $w_1 \models \exists x\phi(x)$ and $w_2 \models \exists x\phi(x)$, so there is a position j in w_2 such that $(w_1,i) \equiv_{k-1} (w_2,j)$. We have already shown that Player II can win the $k-1$-round game in these two structures, so Player II has a winning strategy for the k-round game in w_1 and w_2.

Conversely, suppose w_1 and w_2 are not $\equiv_k$-equivalent. There is thus a two-variable sentence $\exists x\phi(x)$, where ϕ has depth less than k, satisfied by one (say, w_1) and not the other of the two words. Thus Player I can play in w_1 and produce a structure (w_1,i) such that any reply (w_2,j) by Player II is not $(k-1)$-equivalent. Thus, by what we proved above, Player I can win the game in the next $k-1$ rounds.

It follows from this game characterization, and the fact that $\equiv_0$ is a congruence on structures, that $\equiv_k$ is a congruence of finite index on Σ^*. From our normal form result in 4.1, every language defined by a two-variable sentence is a union of $\equiv_k$-classes for some k, m and r. So it is enough to prove that the quotient monoid $\Sigma^*/\equiv_k$ belongs to $\mathbf{DA} * \mathbf{G}_{sol}$.

We will prove this by induction on k. $\Sigma^*/\equiv_0$ is the free Σ-generated group in $\mathbf{H}^r$, and thus is in $\mathbf{G}_{sol}$. The passage from 0 to 1 is a special case: It follows from a result in Straubing [21] that the syntactic monoid of any language defined by a sentence of depth 1 is in $\mathbf{J}_1\Box\mathbf{G}_{sol}$, which (Rhodes and Tilson [15]) is the same as $\mathbf{J}_1 * \mathbf{G}_{sol}$. Since each $\equiv_1$-class is such a language, it follows that $\Sigma^*/\equiv_1 \in \mathbf{J}_1 * \mathbf{G}_{sol}$.

We now carry out the inductive step from $\equiv_k$ to $\equiv_{k+1}$, where $k \geq 1$. We claim that under the homomorphism from $\Sigma^*/\equiv_{k+1}$ to $\Sigma^*/\equiv_k$, the preimage of each idempotent is in $\mathbf{LI}$. Since $\mathbf{LI}^{-1}(\mathbf{LI}^{-1}\mathbf{V}) = \mathbf{LI}^{-1}\mathbf{V}$ for every pseudovariety $\mathbf{V}$ of finite monoids, this will show that the syntactic monoid of each $\equiv_{k+1}$-class is in $\mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{G}_{sol}) = \mathbf{DA} * \mathbf{G}_{sol}$, and thus complete the proof.

Suppose u and v are $\equiv_k$-equivalent words in Σ^*, and are idempotent in $\Sigma^*/\equiv_k$. Suppose further that u is idempotent in $\Sigma^*/\equiv_{k+1}$. We need to show

$$uvu \equiv_{k+1} u.$$

(That's what it means for this to be a $\mathbf{LI}$-morphism: That the inverse image of each idempotent satisfies the identity $ese = e$ whenever e is idempotent.)

Since u is idempotent in $\Sigma^*/\equiv_{k+1}$, the above equation is equivalent to:

$$uuvuu \equiv_{k+1} uuuuu$$

By Lemma 7 it suffices to show that Player II has a winning strategy on this pair of words in the $(k+1)$-round game. The strategy is simply this: If Player I moves anywhere but the middle segment of one of the words Player II will respond on the corresponding position in the other word. If Player I ever moves into the middle segment, Player II will respond according to her strategy for the k-round game in u and v. If Player I moves out of the middle segment and back in again, Player II picks up the middle segment strategy again, starting from the beginning. This strategy will work for Player II unless Player I makes *all* his moves in the middle segments. In that case, after k rounds, the two structures look like this:

$$uu(v,i)uu, uu(u,j)uu.$$

Suppose Player I now moves to the right in the first word, remaining in v, giving

$$uu(v,i')uu$$

with $i < i'$ Player II might not be able to respond in the middle segment of the other word. Instead, she picks a position j' in u such that $(v,i') \equiv_0 (u,j')$ (such a position exists because $u \equiv_k v$ and $k \geq 1$) and moves the pebble to the right to produce

$$uuu(u,j')u.$$

Since $\equiv_0$ is a congruence on word structures, this implies (using the fact that u, being idempotent for $\equiv_k$, is idempotent for $\equiv_0$) that these two structures are $\equiv_0$-equivalent. Thus whatever Player I does, Player II can play safely for $k+1$ successive rounds.

5 Ideal Structure of Monoids in $\mathbf{DA} * \mathbf{G}_{sol}$.

In this section we establish some algebraic properties of pseudovarieties of the form $\mathbf{DA} * \mathbf{H}$, where $\mathbf{H}$ is a pseudovariety of finite groups. Most parts of our Lemma 8 below, as well as Theorem 11 can be extracted from results of Steinberg [19]. Let $\mathcal{F}$ be a set of partial one-to-one functions from a finite set X into itself. We will denote the image of $x \in X$ under $f \in \mathcal{F}$ by xf. We say that $\mathcal{F}$ is $\mathbf{H}$-*extendible* if there is a finite set Y with $X \subseteq Y$, and a permutation group G on Y such that $G \in \mathbf{H}$, and for each $f \in \mathcal{F}$ there exists $g \in G$ such that f is equal to the restriction of g to the domain of f.

Lemma 8. *Let $M \in \mathbf{DA} * \mathbf{H}$, where $\mathbf{H}$ is a pseudovariety of finite groups. Let $\psi : \Sigma^* \to M$ be a homomorphism. Let J be a regular $\mathcal{J}$-class of M, and let (A, B, G, P) be a Rees matrix representation of J.*
(a) There exist a partition of A, a partition of B, and a bijection between the sets of blocks of the two partitions such that $P(b, a) \neq 0$ if and only if the blocks containing a and b correspond under the bijection.
(b) Let $\mathcal{B}$ denote the set of blocks of the partition of B. Let $s \in M$. There is a one-to-one partial function $\pi_s : \mathcal{B} \to \mathcal{B}$ such that if $b \in B$, and $B(b)$ is the block containing b, then $B(b)\pi_s$ is defined if and only if $(a, g, b)s \in J$, in which case $B(b)\pi_s$ is the block containing the right co-ordinate of $(a, g, b)s$. Moreover, the set of partial functions $\{\pi_s : s \in M\}$ is $\mathbf{H}$-extendible.
(c) Let B_1, B_2 be two blocks of the partition of B. Then the language

$$\{w \in \Sigma^* : B_1\pi_{\psi(w)} = B_2\}$$

is recognized by a monoid in $\mathbf{J}_1 * \mathbf{H}$
(d) Suppose $\mathbf{H} * \mathbf{H} = \mathbf{H}$. Let $(a, g, b) \in J$, $g' \in G$. Then the language

$$\{w \in \Sigma^* : (a, g, b)\psi(w) \in \{a\} \times \{g'\} \times B\}$$

is recognized by a monoid in $\mathbf{R} * \mathbf{H}$.

Proof. We first prove the following property (which, in fact, characterizes $\mathbf{DA} * \mathbf{G}$.):
Let $M \in \mathbf{DA} * \mathbf{H}$ and e, f are $\mathcal{J}$-equivalent idempotents of M, and $ef \equiv_{\mathcal{J}} e$, then ef is idempotent.

To show this, we use Lemma 4: $\mathbf{DA} * \mathbf{H} = \mathbf{DA}^{-1}\mathbf{H}$. There is thus a group $H \in \mathbf{H}$ and a $\mathbf{DA}$-relational morphism $\alpha : M \to H$. From the idempotency of e and f it follows that

$$\alpha : e \mapsto 1, \alpha : f \mapsto 1.$$

Since $ef \leq_{\mathcal{R}} e$ and $ef \equiv_{\mathcal{J}} e$, it follows that $ef \equiv_{\mathcal{R}} e$, and thus there is $s \in M$ such that

$$e = efs = effs = ef(fs)^{\omega}.$$

Since $\alpha : (fs)^{\omega} \mapsto 1$, this shows e and ef are $\mathcal{R}$-equivalent, and thus $\mathcal{J}$-equivalent, in $\alpha^{-1}(1)$. Since $\alpha^{-1}(1) \in \mathbf{DA}$, this implies ef is idempotent.

Let us now establish part (a) of the lemma. Consider a Rees matrix representation (A,B,G,P) of a regular $\mathcal{J}$-class J of $M \in \mathbf{DA} * \mathbf{H}$. For each $a \in A$ define

$$B(a) = \{b \in B : P(b,a) \neq 0\}.$$

Now suppose $B(a) \cap B(a') \neq \emptyset$. Then there is some b such that $P(b,a)$ and $P(b,a')$ are nonzero. Suppose further that $P(b',a) \neq 0$ for some $b' \in B$. Thus there are $g, g' \in G$ such that $e = (a', g, b)$ and $f = (a, g', b')$ are idempotent, and $f \in J$. By our claim, ef is idempotent, which implies $P(b', a') \neq 0$. Thus the sets $B(a)$ and $B(a')$ are identical, and so the family of sets $\{B(a) : a \in A\}$ partitions B. We define a partition of A in the analogous fashion. Observe now that if $b, b' \in B(a)$, then $P(b,a)$ and $P(b',a)$ are both nonzero, so that $a \in A(b) \cap A(b')$, and thus $A(b) = A(b')$. This gives the one-to-one correspondence between the sets of blocks of the two partitions, proving part (a) of the lemma.

For part (b), suppose $(a,g,b)s = (a_1, g_1, b_1)$, for some $s \in S$. Since (a_1, g_1, b_1) is $\mathcal{R}$-equivaent to (a,g,b), we have $a_1 = a$. We can write $(a,g,b) = (a,g,b)(\alpha, P(b,\alpha)^{-1}, b)$ for some $\alpha \in A$, and set $t = (\alpha, P(b,\alpha)^{-1}, b)s$. Thus $t = (\alpha, P(b,\alpha)^{-1} g^{-1} g_1, b_1)$. In particular, b_1 depends only on b and s. Thus we have a well-defined partial function ρ_s from B to itself, with $b_1 = \rho_s(b)$. If b' belongs to the same block as b, then for any a', g', we have $(a', g', b')t \in J$, because $P(b', \alpha) \neq 0$, and so $(a', g', b')s \in J$. Thus $\rho_s(b')$ is defined whenever $\rho_s(b)$ is. Let $(a', g', b')s = (a_2, g_2, b_2)$. Choose α' so that $P(b_1, \alpha')$ is nonzero, and let $u = s(\alpha', 1, b)$. Then $(a,g,b)u \in J$, which implies $u \in J$, and thus, since b and b' belong to the same block, $(a', g', b')u \in J$. This implies $P(b_2, \alpha') \neq 0$, and thus $b_1 = \rho_s(b)$ belongs to the same block as $b_2 = \rho_s(b')$.

This shows that there is a well-defined partial function π_s on the set $\mathcal{B}$ of blocks. Suppose now that for some $b, b' \in B$, $\rho_s(b)$ and $\rho_s(b')$ are in the same block. We set $(a,g,b)s = (a_1, g_1, b_1)$ and $(a', g', b')s = (a_2, g_2, b_2)$. Choose α' such that $P(b_1, \alpha')$ is nonzero. Thus $P(b_2, \alpha')$ is nonzero, and so with $u = s(\alpha', 1, b)$ we have $(a,g,b)u \in J$ and $(a', g', b')u \in J$. This implies $u \in J$, and thus if α'' is the left co-ordinate of u, $P(b, \alpha'')$ and $P(b', \alpha'')$ are both nonzero. Thus b and b' are in the same block, and consequently π_s is one-to-one.

It remains to prove the $\mathbf{H}$-extendibility property. We will begin by analyzing the structure of semidirect products $S * H$, where $S \in \mathbf{J}_1$ and $H \in \mathbf{H}$. Idempotents in $S * H$ have the form $(s, 1)$, where $s \in S$. What is the $\mathcal{R}$-class of such an idempotent? If $(s,1) \equiv_{\mathcal{R}} (s', g)$, then there exist $t, t' \in S$ such that

$$(s,1)(t,g) = (s',g),$$

and

$$(s',g)(t', g^{-1}) = (s,1).$$

Thus

$$s' = s + t, s = s' + gt'.$$

Observe that in S, the $\leq_{\mathcal{R}}$, $\leq_{\mathcal{L}}$, and $\leq_{\mathcal{J}}$ orderings all coicide, and the $\mathcal{J}$-classes are singletons. The equations above imply $s' \leq s \leq s'$ (we don't need the subscripts on the $\leq$ symbols) and thus $s = s'$. It follows that the $\mathcal{R}$-class of $(s,1)$ is $\{s\} \times G$.

What is the $\mathcal{L}$-class of $(s,1)$? If $(s',g) \equiv_{\mathcal{L}} (s,1)$ then there exist $t,t' \in S$ such that

$$(t,g)(s,1) = (s',g),$$

and

$$(t',g^{-1})(s',g) = (s,1).$$

Thus

$$s' = t + gs, s = t' + g^{-1}s',$$

so $s' \leq gs$. Also $gs = gt + s'$, from which it follows that $gs \leq s'$, and thus $s' = gs$. So the $\mathcal{L}$-class of $(s,1)$ consists of all elements of the form (gs,g), where $g \in H$.

It follows now that the $\mathcal{J}$-class J of $(s,1)$ consists of all elements of the form (gs,h), where $g,h \in H$. Two such elements, $(g_1s,h_1),(g_2s,h_2)$, are $\mathcal{R}$-equivalent if $g_1s = g_2s$, that is, if g_1 and g_2 belong to the same left coset of the subgroup

$$K(s) = \{g \in H : gs = s\}$$

of H. The two elements are $\mathcal{L}$-equivalent if $h_2h_1^{-1}g_1s = g_2s$, that is, if $h_1^{-1}g_1$ and $h_2^{-1}g_2$ belong to the same left coset of $K(s)$. We can thus identify the sets A and B in the Rees matrix representation of J with the set of left cosets of $K(s)$. Observe that each $\mathcal{L}$-class, as well as each $\mathcal{R}$-class, contains a single idempotent, and thus each block of the partition of B contains a single element. Let us look now at the action of $S * H$ on these blocks. Let $(g_1s,g_2) \in J$ and $(t,h) \in S * H$. Then

$$(g_1s,g_2)(u,h) = (g_1s + g_2u, g_2h).$$

For this product to be in J, we need $g_1s + g_2u = g_1s$. In this case the $\mathcal{L}$-class of (g_1s,g_2) is associated with the left coset $g_2^{-1}g_1K(s)$, and the $\mathcal{L}$-class of the product with the left coset $h^{-1}g_2^{-1}g_1K(s)$. We can thus extend the partial function on blocks induced by (u,h) to the permutation

$$gK(s) \mapsto h^{-1}gK(s)$$

of the set of left cosets. The resulting permutation group is a quotient of H. Thus, for the regular $\mathcal{J}$-classes of $S * H$, the set $\{\pi_{(s,h)} : (s,h) \in S * H\}$ is $\mathbf{H}$-extendible.

To obtain $\mathbf{H}$-extendibility for arbitrary monoids in $\mathbf{DA}*\mathbf{H}$, we apply Lemma 4: $\mathbf{DA} * \mathbf{H} = \mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{H})$. Thus if $M \in \mathbf{DA} * \mathbf{H}$, there is a semidirect product $N = S*H$, with $S \in \mathbf{J}_1$ and $H \in \mathbf{H}$, and an $\mathbf{LI}$-relational morphism $\alpha : M \to N$. Let J be a regular $\mathcal{J}$-class of M, and consider two elements $s = (a,1,b_1)$ and

$t = (a, 1, b_2)$ of J, relative to a fixed Rees matrix representation of this class. We now choose a $\leq_{\mathcal{J}}$-minimal element x of N such that

$$\alpha : s \mapsto x.$$

Since $t = ss'$ for some $s' \in J$, we have, for some $x' \in N$,

$$\alpha : t \mapsto y = xx' \leq_{\mathcal{R}} x.$$

Similarly, $s = tt'$ for some $t' \in J$, so we have, for some $y' \in N$,

$$\alpha : s \mapsto xx'y' \leq_{\mathcal{R}} y \leq_{\mathcal{R}} x.$$

The minimality condition implies that all these elements belong to the same $\mathcal{J}$-class J' of N. Thus $x \equiv_{\mathcal{R}} y$ in N.

Since s is $\mathcal{J}$-equivalent (in fact $\mathcal{R}$-equivalent) to an idempotent, the same sort of minimality argument shows that x is as well, so J' is a regular $\mathcal{J}$-class of N.

We now claim that x and y are $\mathcal{L}$-equivalent in N if and only if b_1 and b_2 belong to the same B-block of J. First, suppose b_1 and b_2 belong to the same block. Then $su, tu \in J$ for some $u \in J$. We can again show, using minimality, that there exists $z \in J'$ such that $\alpha : u \mapsto z$, and hence $xz, yz \in J'$. Thus, in a Rees matrix representation of J', the right co-ordinates of x and y belong to the same block, and are consequently $\mathcal{L}$-equivalent. Conversely, suppose x and y are $\mathcal{L}$-equivalent. Let us choose $u, v \in J$ such that $e = usv$ is an idempotent in J. There exist (again by minimality) $w, z \in J'$ such that $\alpha : u \mapsto w$ and $\alpha : v \mapsto z$. There also exists $n > 0$ such that m^n is idempotent for all $m \in N$. Thus

$$\alpha : e = (usv)^n \mapsto (wxz)^n = f,$$

which is idempotent. By minimality, the idempotent $(wyz)^n$ belongs to the same $\mathcal{R}$-class and $\mathcal{L}$-class as f, and thus $(wyz)^n = f$. It follows that both e and $(utv)^n$ belong to $\alpha^{-1}(1)$. Since α is an **LI**-relational morphism,

$$e = e(utv)^n e = e \in J.$$

This shows that tv, like sv, is in J, and thus b_1, b_2 belong to the same B-block of J.

It follows that α embeds the set of B-blocks of J into the set of B-blocks of J'. The action of π_m, for $m \in M$, on the B-blocks of J, is identical to the action of π_n, where $\alpha : m \mapsto n$, on the B-blocks of J'. Since we have **H**-extendibility for N, we now have it for M as well.

To prove part (c), it is sufficient to show that the language is recognized by an automaton whose underlying transformation monoid is in $\mathbf{J}_1 * \mathbf{H}$. By what we showed above, there is a superset X of the set of B-blocks of J and a permutation group H on X such that $H \in \mathbf{H}$, and for each $s \in M$ there exists $\pi'_s \in H$ that extends π_s. The state set of the automaton is $\{0, 1\} \times X$, the initial state is $(1, B_1)$ and the sole accepting state is $(1, B_2)$. The action of $\sigma \in \Sigma$ on the state set is given by

$$(i, x)\sigma = (i \cdot j, x \cdot \pi'_{\psi(\sigma)}),$$

where $j = 1$ if x is in the domain of $\pi_{\psi(\sigma)}$, and $j = 0$, otherwise. Thus each state transition is a transformation in the wreath product $(\{0,1\}, U_1) \circ (X, H)$, where U_1 is the monoid $\{0,1\}$ with the standard multiplication. Since $U_1 \in \mathbf{J}_1$, this establishes the claim.

To prove part (d), it is helpful to have the following normal form for the Rees matrix representation of a regular $\mathcal{J}$-class: We claim that the representation can be chosen so that for all $b \in B$, $a \in A$, $P(b,a)$ is either 0 or 1. To prove this, we choose a fixed representative for each block in the partition of A, and for each block in the partition of B: If $a \in A$, $b \in B$, then $\alpha(a)$, $\alpha(b)$, $\beta(a)$, $\beta(b)$ denote the chosen representatives of $A(a)$, $A(b)$, $B(a)$, $B(b)$, respectively.

Let $e_i = (a_i, g_i, b_i)$, $i = 1, 2$, be idempotents with $P(b_1, a_2) \neq 0$. Idempotence implies $g_i = P(b_i, a_i)^{-1}$ for $i = 1, 2$. We have $e_1 e_2 \in J$, so as we proved above, $e_1 e_2$ is idempotent. If we write the equation $e_1 e_2 = e_1 e_2 e_1 e_2$ in terms of the Rees matrix representation, we obtain:

$$P(b_2, a_1) P(b_1, a_1)^{-1} P(b_1, a_2) P(b_2, a_2)^{-1} = 1$$

with this identity holding whenever $a_2 \in A(a_1)$ and $b_1, b_2 \in B(a_1)$. We now define a new map $P^* : B \times A \to G \cup \{0\}$ by setting $P^*(b,a) = P(\beta(a), \alpha(b))^{-1}$ if $P(b,a) \neq 0$, and $P^*(b,a) = 0$ otherwise. A straightforward computation, using the above identity, shows that the map

$$(a, g, b) \mapsto (a, P(\beta(a), a)^{-1} g P(b, \alpha(b))^{-1}, b)$$

is an isomorphism from (A, B, G, P^*) onto (A, B, G, P).

We now define a map $Q : B \times A \to G \cup \{0\}$ by setting $Q(b,a)$ to be 1 if $P(b,a) \neq 0$ and $Q(b,a) = 0$ otherwise. The map

$$(a, g, b) \mapsto (a, g P^*(\beta(b) \alpha(b))^{-1}, b)$$

then defines an isomorphism from (A, B, G, Q) onto (A, B, G, P^*).

We now suppose that the Rees matrix representation of our $\mathcal{J}$-class J has this normal form. Suppose that $(a, g, b) \in J$, $s \in M$, and $(a, g, b)s = (a, g', b') \in J$. We claim that for every $(a_1, g_1, b_1) \in J$ with b_1 in the same block as b, $(a_1, g_1, b_1)s = (a_1, g_1 g^{-1} g', \rho_s(b_1))$. To see this, note that

$$(a, g, b) s (\alpha(b'), 1, b') = (a, g', b'),$$

so

$$s(\alpha(b'), 1, b') = (a', g^{-1} g', b'),$$

where $a' \in A(b)$. Thus

$$(a_1, g_1, b_1) s (\alpha(b'), a, b) = (a_1, g_1 g^{-1} g', b'),$$

which proves the claim. Therefore there is a map f_s from the set of B-blocks of J into G such that whenever $(a, g, b)s \in J$, the second co-ordinate of $(a, g, b)s$ is $g \cdot f_s(B(b))$. We use this fact to construct an automaton that recognizes the given language. The state set of the automaton is $G \times \{0,1\} \times X$, where X is as defined in the first part of the proof. The initial state is $(g, 1, B(b))$ and the set of final states is $\{g'\} \times \{1\} \times X$. The action of $\sigma \in \Sigma$ on the state set is given by

$$(h, i, x)\sigma = (h \cdot f_{\psi(\sigma)}(x), i \cdot j, x\pi'_{\psi(\sigma)}),$$

where j is as in the first part of this proof. (The map $f_{\psi(\sigma)}$ can be extended to X in an arbitrary manner.) Each state transition is a transformation in the wreath product

$$(G, G) \circ (\{0, 1\}, U_1) \circ (X, H).$$

Since G, being a group contained in M, must itself belong to $\mathbf{H}$, it follows from results in the cited reference by Stiffler, and from our assumption that $\mathbf{H}*\mathbf{H} = \mathbf{H}$, that the underlying monoid of this wreath product is in $\mathbf{R} * \mathbf{G}_{sol}$.

6 Two-Variable Definability for $\mathbf{DA} * \mathbf{G}_{sol}$.

Let Σ be a finite alphabet. We will prove in this section that if $L \subseteq \Sigma^*$ is recognized by a monoid $M \in \mathbf{DA} * \mathbf{G}_{sol}$, then L is definable by a sentence with two variables. This will complete the proof of Theorem 3.

Let $\phi : \Sigma^* \to M$ be a homomorphism. Each $w \in \Sigma^*$ has a unique factorization

$$w = w_0\sigma_1 w_1 \cdots \sigma_k w_k,$$

where each σ_i is in Σ, $\phi(w_0) \equiv_{\mathcal{R}} 1$, and where, for $i = 1, \ldots, k$,

$$\phi(w_0\sigma_1 \cdots \sigma_i w_i) \equiv_{\mathcal{R}} \phi(w_0\sigma_1 \cdots w_{i-1}\sigma_i) <_{\mathcal{R}} \phi(w_0\sigma_1 \cdots w_{i-1}).$$

Let $s, t \in M$, with $s \equiv_{\mathcal{R}} t$. We define $L[s, t] = \{w \in \Sigma^* : s \cdot \phi(w) = t\}$. Thus, if $m \in M$, $\phi^{-1}(m)$ is the union of all languages of the form

$$L[1, t_0]\sigma_1 L[t_0 \cdot \phi(\sigma_1), t_1] \cdots \sigma_k L[t_{k-1} \cdot \phi(\sigma_k), t_k], \tag{1}$$

where $t_k = m$, and, for $i = 1, \ldots, k$, $t_i \equiv_{\mathcal{R}} t_{i-1} \cdot \phi(\sigma_i) <_{\mathcal{R}} t_{i-1}$. This union is finite, since k is bounded above by the number of $\mathcal{R}$-classes of M. It is therefore sufficient to show that every language of the form (1) is definable by a two-variable sentence. We prove this by induction on $|M|$: If $|M| = 1$, then the language (1) is Σ^*, which is defined by the one-variable sentence $\neg\exists x(x < x)$. We thus suppose $|M| > 1$. Our inductive hypothesis is that for all $M' \in \mathbf{DA} * \mathbf{G}_{sol}$ with $|M'| < |M|$, languages of the form (1) are two-variable definable.

We prove the assertion for M by a second induction, this time on k. We begin by considering languages of the form $L[s, t]$. First, suppose that the $\mathcal{R}$-class containing s and t is contained in a regular $\mathcal{J}$-class J of M. We identify J with a Rees matrix representation (A, B, G, P). There is a partition of B as specified in Lemma 8; as before, we denote by $B(b)$ the block of this partition containing the element b. Let $s = (a, g, b), t = (a'g', b')$. In order for a word w to belong to $L[s, t]$, we need either:

(a) $\phi(w) \notin J$ and $s\phi(w) = t$, or

(b) $\phi(w) \in J$, $B(b)$ is in the domain of $\pi_{\phi(w)}$, the middle co-ordinate of $(a, g, b)\phi(w)$ is g', and $w = w_1\sigma w_2$, where $\sigma \in \Sigma$, $\phi(w_2) \notin J$, and $\phi(\sigma w_2) \in J$ with right co-ordinate b'.

The set of strings satisfying condition *(a)* is recognized by the monoid M/I, where I is the ideal consisting of all elements of M that are not strictly above J in the $\mathcal{J}$-ordering. If $|M/I| = |M|$ then J consists of a single element, which is the zero of M, and $L[s,t] = \Sigma^*$. Thus we may suppose $|M/I| < |M|$. Since $M/I \in \mathbf{DA} * \mathbf{G}_{sol}$, the inductive hypothesis implies that this set of strings is two-variable definable.

By Lemma 8, the set of strings w such that $B(b)$ is in the domain of $\pi_{\phi(w)}$ is recognized by a monoid in $\mathbf{J}_1 * \mathbf{G}_{sol}$, and, since $\mathbf{G}_{sol} * \mathbf{G}_{sol} = \mathbf{G}_{sol}$, the set of strings w such that the middle co-ordinate of $(a, g, b)\phi(w)$ is g' is recognized by a monoid in $\mathbf{R} * \mathbf{G}_{sol}$. By Theorem 6, these are both definable by left-relativizable two-variable sentences. Let $\sigma \in \Sigma$, and let K_σ be the set of strings $w_1\sigma w_2$, where $\phi(w_2) \notin J$, and $\phi(\sigma w_2) \in J$ with right-co-ordinate b. K_σ is then a union of sets of the form (1), but with the sets $L[s,t]$ replaced by their $\mathcal{L}$-class duals, with respect to the monoid M/I. Since, as we noted in the remarks following Lemma 4, $\mathbf{DA} * \mathbf{G}_{sol}$ is closed under reversal, the inductive hypothesis implies that each K_σ is two-variable definable.

In the case where J is a null $\mathcal{J}$-class, the product of two elements of J is not in J. Thus $L[s,t]$ is recognized by M/I, where I is as defined above, and is thus two-variable definable.

We now suppose that we have a two-variable sentence δ for the language

$$L[t_i\phi(\sigma_{i+1}), t_{i+1}]\sigma_{i+2} \cdots L[t_{k-1}\phi(\sigma_k), t_k],$$

and use it to obtain a two-variable definition for $L' = L[t_{i-1}\phi(\sigma_i), t_i]\sigma_{i+1}L$. First we consider the case where the $\mathcal{J}$-class J that contains $t_{i-1}\phi(\sigma_i)$ and t_i is regular. Let $t_{i-1}\phi(\sigma_i) = (a_1, g_1, b_1)$, $t_i = (a_2, g_2, b_2)$. Let θ be a left-relativizable two-variable sentence for the set of strings u such that $B(b_1)$ is in the domain of $\pi_{\phi(u)}$, and η a left-relativizable two-variable sentence for the set of strings u such that the middle co-ordinate of $(a_1, g_1, b_1)\phi(u)$ is g_2. Such sentences exist by Lemma 8 and Theorem 6.

Let ζ be a two-variable sentence for the set of strings u such that $u = u_1\sigma u_2$, with $\sigma \in \Sigma$, $\phi(u_2) \notin J$, and $\phi(\sigma u_2) \in J$ with right co-ordinate b_2. We showed above that such a sentence exists. Let ζ' be a two-variable sentence for the set of strings u such that $\phi(u) \notin J$, and $(a_1, g_1, b_1)\phi(u) = (a_2, g_2, b_2)$. Again, we showed above that such a sentence exists.

Our sentence defining L' is

$$\exists x(Q_{\sigma_{i+1}}x \wedge \theta[< x] \wedge \neg\theta[\leq x]) \wedge ((\eta \wedge \zeta) \vee \zeta')[\leq \theta] \wedge \delta[> \theta].$$

Observe that because of the left-relativizability of θ, all the relativizations in the above sentence are two-variable formulas.

For the case of a null $\mathcal{J}$-class, we proceed in the identical fashion, except now we do not need formulas analogous to η and ζ.

7 Extensions of the Main Result and Directions for Further Research

7.1 Characterization of DA.

As we mentioned in the Introduction, Thérien and Wilke [26] considered two-variable sentences with only ordinary quantifiers and proved:

Theorem 9. *L is definable by a first-order sentence with two variables if and only if $M(L) \in \mathbf{DA}$.*

We have not used this result in the proof of our main theorem; in fact, it follows fairly easily from the arguments that we used to prove Theorem 3. Let us briefly describe how. In the case of two-variable formulas without modular quantifiers, the equivalence relation $\equiv_0$ on words, introduced in 4.2, is the trivial equivalence that identifies all words, and the relation $\equiv_0$ on structures identifies (w_1, i) and (w_2, j) if the i^{th} letter of w_1 is equal to the j^{th} letter of w_2. Lemma 7 continues to hold. It is easy to see that for words w_1, w_2, $w_1 \equiv_1 w_2$ if and only if w_1 and w_2 contain the same letters, and thus the syntactic monoid of any language defined by a sentence of depth 1 is in $\mathbf{J}_1$. The game argument of 4.2 then shows that the syntactic monoid of every language definable by a two-variable sentence is in $\mathbf{LI}^{-1}(\mathbf{J}_1) = \mathbf{DA}$. Observe that the normal form result in 4.1 is of no relevance.

For the converse, we observe that for regular $\mathcal{J}$-classes of monoids in $\mathbf{DA}$, the partitions of A and B into blocks are trivial: There is just one A-block and one B-block. The argument given in Section 6 now goes through as before; we need only observe that we never have to introduce a modular quantifier. Note that the set of strings w such that the unique B-block is in the domain of $\pi_{\phi(w)}$ therefore has the form Γ^*, where $\Gamma \subseteq \Sigma$, and is thus defined by the one-variable sentence $\vee_{\gamma \in \Gamma} Q_\gamma x$. We need never concern ourselves with the middle coordinates in Rees matrix representations, so we do not require a sentence analogous to the sentence η in that proof.

7.2 $\boldsymbol{\Sigma_2[MOD] \cap \Pi_2[MOD]}$

Let us denote by $\Pi_2[<]$ the family of languages over Σ defined by Π_2-sentences over the base of atomic formulas $x_i < x_j$, and $Q_\sigma x_i$, where $\sigma \in \Sigma$. Note that there is no restriction on the number of variables. Similarly, we denote by $\Sigma_2[<]$ the family of languages defined by Σ_2-sentences. Pin and Weil [14] prove that $\Sigma_2[<] \cap \Pi_2[<]$ is exactly the family of languages whose syntactic monoids belong to the pseudovariety $\mathbf{DA}$.

Here we extend this theorem: Let us denote by $\Pi_2[MOD]$ the family of languages defined by Π_2-sentences over the base of formulas that use only the aforementioned atomic formulas and modular quantifiers. $\Sigma_2[MOD]$ is defined analogously. Here we prove that the intersection of these two classes is exactly the class of languages definable with two variables.

Theorem 10. *Let $L \subseteq \Sigma^*$. $L \in \Sigma_2[MOD] \cap \Pi_2[MOD]$ if and only if $M(L) \in$ $\mathbf{DA} * \mathbf{G}_{sol}$.*

Proof. We first suppose $M(L) \in \mathbf{DA}*\mathbf{G}_{sol}$ and prove $L \in \Sigma_2[MOD] \cap \Pi_2[MOD]$. By Lemma 4,

$$\mathbf{DA} * \mathbf{G}_{sol} = \mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{G}_{sol}).$$

It follows now from results of Pin, Straubing and Thérien [13], and Pin and Weil [14], that

$$L \in \mathrm{Pol}(\mathbf{J}_1 * \mathbf{G}_{sol}) \cap \mathrm{CoPol}(\mathbf{J}_1 * \mathbf{G}_{sol}).$$

Here, $\mathrm{Pol}(\mathbf{V})$ denotes the family of languages that are unions of concatenation products of the form

$$L_0\sigma_1 L_1 \cdots \sigma_k L_k,$$

where for all i, $\sigma_i \in \Sigma$, and $M(L_i) \in \mathbf{V}$. $\mathrm{CoPol}(\mathbf{V})$ denotes the family of complements of these languages. (See Theorems 7.1 and 7.2 of [14].)

Since $\Sigma_2[MOD]$ is closed under union, and since the complement of a language in $\Sigma_2[MOD]$ is in $\Pi_2[MOD]$, it suffices to show that in the case $\mathbf{V} = \mathbf{J}_1 * \mathbf{G}_{sol}$, each such concatenation product belongs to $\Sigma_2[MOD]$.

The languages L_i whose syntactic monoids belong to $\mathbf{J}_1 * \mathbf{G}_{sol}$ are boolean combinations of languages of the form $K\sigma\Sigma^*$, where $M(K) \in \mathbf{G}_{sol}$, and $\sigma \in \Sigma$. These are consequently definable by boolean combinations of sentences of the form $\exists x\phi(x)$, where ϕ uses only modular quantifiers. It follows that the product $L_0\sigma_1 L_1 \cdots \sigma_k L_k$ is defined by a sentence of the form

$$\exists x_1 \exists x_2 \cdots \exists x_k \psi(x_1, \ldots, x_k),$$

where ψ is a boolean combination of formulas of the form

$$\exists x(x < x_i \wedge x < x_{i+1} \wedge Q_{\sigma_i}(x_i) \wedge Q_{\sigma_{i+1}}(x_{i+1}) \wedge \phi'(x, x_i, x_{i+1})),$$

and where ϕ' is the relativization of ϕ to the interval between x_i and x_{i+1}. (We mean to include in this general description the formulas in which only x_k is free and in which only x_1 is free.) We can write this boolean combination ψ in disjunctive normal form. Now the quantifier block

$$\exists x_1 \cdots \exists x_k$$

commutes with $\vee$, and

$$\exists z\alpha \wedge \exists z\beta$$

is equivalent to

$$\exists z' \exists z''(\alpha' \wedge \beta''),$$

where α' and β'' are obtained from α and β by renaming variables. It follows that our sentence is equivalent to a disjunction of sentences of the form

$$\exists x_1 \cdots \exists x_k (\exists z_1 \cdots \exists z_r \gamma \wedge \neg\exists y\delta),$$

where γ and δ use only modular quantifiers. Each such sentence is in turn equivalent to

$$\exists x_1 \cdots \exists x_k \exists z_1 \wedge \exists z_r \forall y (\gamma \wedge \neg\delta).$$

Since the disjunction of Σ_2-sentences is a Σ_2-sentence, we have $L \in \Sigma_2[MOD]$, as required.

For the converse direction, let $L \in \Sigma_2[MOD] \cap \Pi_2[MOD]$. Since $L \in \Sigma_2[MOD]$, L is defined by a sentence of the form

$$\exists x_1 \cdots \exists x_k \forall y_1 \cdots \forall y_r \psi,$$

where ψ uses only modular quantifiers. Let d be the depth of nesting of the modular quantifiers in ψ, and let m be the least common multiple of the moduli in ψ. Let $\equiv$ be the equivalence relation on Σ^* that identifies two words if they satisfy the same sentences of quantifier depth d that use only modular quantifiers whose moduli divide m. Observe that $\equiv$ is precisely the congruence $\equiv_0$ for modulus m and depth d introduced in 4.2. Let $s > 0$, and let $\cong_s$ be the equivalence relation on Σ^* that identifies two words v and v' if for every factorization

$$v = v_0 \tau_1 v_1 \cdots \tau_p v_p,$$

with $p \leq s$ and $\tau_i \in \Sigma$ for $1 \leq i \leq p$, there exists a factoreization

$$v' = v'_0 \tau_1 v'_1 \cdots \tau_p v'_p,$$

with $v_i \equiv v'_i$ for $0 \leq i \leq p$, and if, conversely, for each such factorization of v' there exists a corresponding factorization of v.

It is not hard to show that $\cong_s$ is a congruence of finite index on Σ^*. (See, for example, Thérien [25].)

We claim that L is a union of languages of the form

$$L_0 \sigma_1 L_1 \cdots \sigma_q L_q,$$

for some q, where each σ_i is a letter of Σ, and each L_i is an $\cong_r$-class. To prove this claim, let $w \in L$. Then

$$w \models \exists x_1 \cdots \exists x_k \forall y_1 \cdots \forall y_r \psi.$$

There thus exists a map $I : \{x_1, \ldots, x_r\} \to \{1, \ldots, |w|\}$ such that

$$(w, I) \models \forall y_1 \cdots \forall y_r \psi.$$

The map I gives a factorization

$$w = w_0 \sigma_1 w_1 \cdots \sigma_t w_t,$$

where the σ_i are the letters in the positions corresponding to the image of I. Note that $t \leq k$. Now let

$$w' = w'_0 \sigma_1 w'_1 \cdots \sigma_t w'_t,$$

where $w_i \cong_r w_i'$ for $1 \leq i \leq t$. Define $I' : \{x_1, \ldots, x_k\} \to \{1, \ldots, |w'|\}$ as follows: If $I(x_j)$ is the position in w occupied by σ_i in the above factorization of w, then $I'(x_j)$ is the position in w' occupied by σ_i in the given factorization of w'. We claim

$$(w', I') \models \forall y_1 \cdots \forall y_r \psi.$$

If not, there is a map $J' : \{y_1, \ldots, y_r\} \to \{1, \ldots, |w'|\}$ such that

$$(w', I' \cup J') \not\models \psi.$$

The elements of the image of J' that correspond to positions within w_i' induce a factorization of w_i'. Since $w_i \equiv_s w_i'$ for each i, there also exists a factorization of w_i such that the corresponding factors are $\cong_r$-equivalent. These factorizations of the w_i, together with the map J', serve to define a map $J : \{y_1, \ldots, y_r\} \to \{1, \ldots, |w|\}$. Because of the $\equiv$-equivalence of the factors, and the multiplicative property of $\equiv$ (noted earlier in 4.2) the resulting structure $(w, I \cup J)$ satisfies all the same modular formulas of quantifier depth d and modulus dividing m as $(w', I' \cup J')$, and therefore $(w, I \cup J)$ does not satisfy ψ, a contradiction. We thus have

$$w' \models \exists x_1 \cdots \exists x_k \forall y_1 \cdots \forall y_r \psi,$$

and so L is a union of products, as claimed.

We denote by $\Diamond \mathbf{G}_{sol}$ the pseudovariety generated by the syntactic monoids of concatenations of regular languages whose syntactic monoids are solvable groups. Each $\cong_r$-class is a regular language whose syntactic monoid belongs to $\Diamond \mathbf{G}_{sol}$. From results of Margolis and Pin (see, for example, Pin [12]) we have

$$\Diamond \mathbf{G}_{sol} = \mathbf{J} * \mathbf{G}_{sol} \subseteq \mathbf{DA} * \mathbf{G}_{sol},$$

and thus $L \in \mathrm{Pol}(\mathbf{DA} * \mathbf{G}_{sol})$. But since we also have $L \in \Pi_2[MOD]$, we have

$$L \in \mathrm{Pol}(\mathbf{DA} * \mathbf{G}_{sol}) \cap \mathrm{CoPol}(\mathbf{DA} * \mathbf{G}_{sol}).$$

It now follows (using results of Pin, Straubing and Thérien [13] and Pin and Weil [14] cited earler, and Lemma 4) that

$$\begin{aligned} M(L) &\in \mathbf{LI}^{-1}(\mathbf{DA} * \mathbf{G}_{sol}) \\ &= \mathbf{LI}^{-1}(\mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{G}_{sol})) \\ &= \mathbf{LI}^{-1}(\mathbf{J}_1 * \mathbf{G}_{sol}) \\ &= \mathbf{DA} * \mathbf{G}_{sol}. \end{aligned}$$

7.3 $\mathbf{G}_{sol}$-extendibility

The biggest question left unanswered by our work is whether one can effectively determine if a given regular language is definable by a two-variable sentence. It follows from our arguments that $L \subseteq \Sigma^*$ is two-variable definable if and only if for every regular $\mathcal{J}$-class J of $M(L)$, J admits a block partition of the kind described in Section 5, and the set $\{\pi_s : s \in M\}$ of partial one-to-one transformations on the set of B-blocks of $\mathcal{J}$ is $\mathbf{G}_{sol}$-extendible. In fact, we are able to prove:

Theorem 11. *The following two decision problems are equivalent: (a) To determine whether a given regular language is two-variable definable. (b) To determine whether a given set of partial one-to-one functions on a finite set is $\mathbf{G}_{sol}$-extendible.*

Proof. First suppose that we have an algorithm to determine whether a given set of partial one-to-one maps of a finite set X to itself is $\mathbf{G}_{sol}$-extendible. Let L be a regular language, given by a finite automaton that recognizes it. We can effectively compute the multiplication table of the syntactic monoid M of L. We can also calculate the regular $\mathcal{J}$-classes M, and determine whether they satisfy the condition that we found in the proof of Lemma 8: If e, f are idempotents in $\mathcal{J}$, and $ef \in \mathcal{J}$, then ef is idempotent. If some regular $\mathcal{J}$-class fails to satisfy this condition, then $M \notin \mathbf{DA} * \mathbf{G}$, and hence, by Theorem 3, L is not two-variable definable. Otherwise, the partitions associated with each regular $\mathcal{J}$-class, as described in Lemma 8, exist, and we can calculate them and the associated sets of partial injective functions. We now apply our algorithm for $\mathbf{G}_{sol}$-extendibility: If one of these sets fails to be extendible, then $M \notin \mathbf{DA}*\mathbf{G}_{sol}$, so by Theorem 3, L is not two-variable definable. Otherwise, we have all the properties we needed to prove parts (c) and (d) of Lemma 8 and to carry through the argument in Section 6, so we conclude that L is two-variable definable.

For the converse, suppose that we have an algorithm to decide whether a given regular language is two-variable definable. It follow from Theorem 3 and standard techniques that there is then an algorithm to test whether a given finite monoid is in $\mathbf{DA} * \mathbf{G}_{sol}$. We will now use this algorithm to test whether a given set $\mathcal{F}$ of partial one-to-one maps on a finite set X is $\mathbf{G}_{sol}$-extendible.

We associate to $\mathcal{F}$ the directed graph with vertex set X and edge set $\{(x, xf) : x \in dom(f)\}$. We may assume without loss of generality that this graph is connected in the undirected sense: Easily, $\mathcal{F}$ is extendible to a solvable permutation group if and only if the restriction of $\mathcal{F}$ to each of its connected components is extendible.

Let $< \mathcal{F} >$ denote the monoid of partial one-to-one functions on X generated by $\mathcal{F}$: $\mathcal{F}$ is extendible to a solvable permutation group G if and only if $< \mathcal{F} >$, considered as a set of partial one-to-one maps, is extendible to G.

We now define a new monoid $M(\mathcal{F})$. The underlying set of $M(\mathcal{F})$ is the disjoint union of $< \mathcal{F} >$ with the Rees matrix semigroup $(X, X, \{1\}, I)$, where $\{1\}$ denotes the trivial group, and I the $|X| \times |X|$ idenity matrix. Both $< \mathcal{F} >$ and $(X, X, \{1\}, I)$ embed as subsemigroups of $M(\mathcal{F})$; it remains to define the product of an element of one of these subsemigroups with an element of the other: If $f \in M(\mathcal{F})$ and $(x, 1, y) \in (X, X, \{1\}, I)$, then

$$(x, 1, y)f = (x, 1, yf),$$

if yf is defined, and 0 otherwise;

$$f(x, 1, y) = (xf^{-1}, 1, y)$$

if xf^{-1} is defined, and 0 otherwise. It is straightforward to verify that this multiplication is associative.

We claim that $\mathcal{F}$ is $\mathbf{G}_{sol}$-extendible if and only if $M(\mathcal{F}) \in \mathbf{DA} * \mathbf{G}_{sol}$. First, suppose that $\mathcal{F}$ is extendible to a solvable permutation group G. The nonzero elements of $(X, X, \{1\}, I)$ for a regular $\mathcal{J}$-class J of $M(\mathcal{F})$. The associated set $\mathcal{F}'$ of partial one-to-one maps on the set of B-blocks of J consists of the elements of $\mathcal{F}$, together with the partial maps f_{xy}, $x, y \in X$, where

$$x' f_{xy} = y,$$

if $x = x'$, and where f_{xy} is undefined otherwise. Our condition on the connectedness of the graph associated with $\mathcal{F}$ implies that $X \subseteq XG$. Thus every map in f_{xy} is extendible to a permutation in G, and so $\mathcal{F}'$ is extendible to G as well. What about the other $\mathcal{J}$-classes of $M(\mathcal{F})$? They all lie in the submonoid $< \mathcal{F} >$. We showed in our proof of part (c) of Lemma 8 that $M(\mathcal{F}) \in \mathbf{J}_1 * \mathbf{G}_{sol}$, and hence by part (b) of the same lemma, each of these $\mathcal{J}$-classes is $\mathbf{G}_{sol}$-extendible. Since every $\mathcal{J}$-class of $M(\mathcal{F})$ is $\mathbf{G}_{sol}$-extendible, our arguments in Section 6 show that every language recognized by $M(\mathcal{F})$ is two-variable definable, and hence $M(\mathcal{F}) \in \mathbf{DA} * \mathbf{G}_{sol}$.

Conversely, suppose $M(\mathcal{F}) \in \mathbf{DA} * \mathbf{G}_{sol}$. By part (b) of Lemma 8, the set of partial one-to-one maps on the blocks of the regular $\mathcal{J}$-class J is $\mathbf{G}_{sol}$-extendible. Since $\mathcal{F}$ is contained in this set of maps, $\mathcal{F}$ is $\mathbf{G}_{sol}$-extendible as well.

There is no algorithm presently known for determining if a given set $\mathcal{F}$ of partial one-to-one maps on a finite set X is $\mathbf{G}_{sol}$-extendible. It is tempting to conjecture that $\mathcal{F}$ is extendible if and only if the monoid of partial one-to-one maps generated by $\mathcal{F}$ contains only solvable groups. We suspect this conjecture is false. To disprove it, one would have to exhibit a solvable monoid of partial one-to-one maps on a finite set and show that it cannot be extended to a solvable permutation group. While it is easy to come up with candidate counterexamples, we do not yet know how to prove non-extendibility.

Margolis, Sapir and Weil [10] show that if $\mathbf{H}$ is a pseudovariety of groups such that $\mathbf{H} * \mathbf{H} = \mathbf{H}$ ($\mathbf{G}_{sol}$ has this property) then the question of $\mathbf{H}$-extendibility is equivalent to the problem of computing the closure of a finitely-generated subgroup of the free group in the profinite topology induced by $\mathbf{H}$. Ribes and Zaleskii [16] showed that this problem is decidable for the pseudovariety $\mathbf{G}_p$ of p-groups for a fixed prime p. As a consequence we have

Theorem 12. *Let p be prime. It is decidable whether a given regular language is definable by a two-variable sentence in which all the modular quantifiers are of modulus p.*

Proof. There is really nothing new to prove here—we simply note that we could redo our entire argument using the pseudovariety $\mathbf{G}_p$ of p-groups in place of the solvable groups. The crucial facts are that sentences whose only modular quantifiers are of modulus p define the languages whose syntactic monoids contain only groups in $\mathbf{G}_p$ (see [24]), and that $\mathbf{G}_p * \mathbf{G}_p = \mathbf{G}_p$.

7.4 Connections with computational complexity

We suspect that our results have a connection to computational complexity theory, in particular to the structure of classes within the polynomial-time hierarchy, and within polynomial space.

Let $\mathcal{C}$ be a class of languages over a finite alphabet Σ. (In discussions of computational complexity, we usually take $\Sigma = \{0, 1\}$.) We define several operators on language classes: Let $\mathcal{Q}$ be a quantifier, either existential, universal, or modular. We define $\mathcal{Q} \cdot \mathcal{C}$ to be the class of languages L for which there exist a polynomial p and a language $K \in \mathcal{C}$ such that

$$w \in L \Leftrightarrow \mathcal{Q}x(x \in \Sigma^{p(|w|)} \wedge wx \in K).$$

We define the class $BP \cdot \mathcal{C}$ to be the class of languages for which there exist a polynomial p and language $K \in \mathcal{C}$ as above, and such that

$$\frac{|\{x \in \Sigma^{p(|w|)} : w \in L \Leftrightarrow wx \in K\}|}{|\Sigma^{p(|w|)}|} \geq \frac{2}{3}.$$

Let $\mathcal{P}$ be the class of polynomial-time recognizable languages. With this notation, $\exists \cdot \mathcal{P}$ is the class $\mathcal{NP}$, and $\forall \cdot \mathcal{P}$ is the class $\text{co} - \mathcal{NP}$. The class

$$\bigcup_{k \geq 0} (\exists \cdot \forall)^k \cdot \mathcal{P}$$

is the polynomial-time hierarchy $\mathcal{PH}$. The class $\exists^{1 \bmod 2} \cdot \mathcal{P}$ is more commonly written $\oplus \mathcal{P}$.

There is a notion of polynomial-time recognizability of languages by finite monoids, which we now describe. (See Hertrampf, et. al. [5] for an an account of the equivalent notion of "leaf languages".) Let M be a finite monoid, $X \subseteq M$, p a polynomial, and $f : \Sigma^* \to M$ a polynomial-time computable function. We say (M, X, p, f) recognizes the language

$$\{w \in \Sigma^* : (\prod_{|x|=p(|w|)} f(wx)) \in X\},$$

where the order of multiplication in the above product is the lexicographic order on $\Sigma^{p(|w|)}$. We also say in this case that this language is polynomial-time recognized by M. It is known, for example, that $L \in$ PSPACE if and only if L is polynomial-time recognized by some finite monoid, and that $L \in \mathcal{PH}$ if and only if L is recognized in this sense by a finite aperiodic monoid.

Toda [29] showed
$$\mathcal{PH} \subseteq BP \cdot \oplus\mathcal{P}.$$
It then follows, using results of Schöning [17], that
$$\mathcal{PH} \subseteq \forall \cdot \exists \cdot \oplus\mathcal{P} \cap \exists \cdot \forall \cdot \oplus\mathcal{P}.$$
This last result can be interpreted algebraically: There is a *fixed* finite aperiodic monoid A such that every language polynomial-time recognized by a finite aperiodic monoid is recognized by $A \circ \mathbf{Z}_2$. (See Straubing [22] for an account of this based on circuit complexity.) There is nothing special about the modulus 2 here— any modulus greater than 1 would work just as well.

There are some striking similarities between these results (and their proofs) and our work on $\mathbf{DA} * \mathbf{G}_{sol}$. First, both illustrate the unexpected power of modular counting to efficiently perform computations that can be done, more cumbersomely, without modular counting. Second, the complexity class
$$\forall \cdot \exists \cdot \oplus\mathcal{P} \cap \exists \cdot \forall \cdot \oplus\mathcal{P}$$
bears an obvious resemblance to the class $\Sigma_2[MOD] \cap \Pi_2[MOD]$ that we discussed in 7.2. Moreover, Toda's theorem is proved, in part, by demonstrating that the modular quantifier operator can be moved past the BP operator, much as we showed in 4.1 that modular quantifiers can be moved past ordinary quantifiers in two-variable formulas. We would like to find the algebraic principles underlying these similarities. We make the following conjecture:
Conjecture. If $L \in \mathcal{PH}$ then L is polynomial-time recognized by a monoid in $\mathbf{DA} * \mathbf{G}_{sol}$.

In fact, the same intuition suggests that this conclusion holds whenever L is polynomial-time recognized by a finite solvable monoid.
Acknowledgements. We would like to thank Stuart Margolis for his very helpful comments, and Benjamin Steinberg for pointing out the reference [19]. The second author's research was supported by grants from NSERC and FCAR, and by the von Humboldt Foundation.

References

1. J. Almeida, "A Syntactical Proof of the Locality of **DA**", *International Journal of Algebra and Computation* **6** (1996) 165-178.
2. D. Beauquier and J. E. Pin, "Factors of Words", *Proc. 16th ICALP,* Springer Lecture Notes in Computer Science **372** (1989) 63–79.
3. S. Eilenberg, *Automata, Languages and Machines,* vol. B, Academic Press, New York, 1976.
4. K. Etessami, M. Vardi, and T. Wilke, "First-Order Logic with Two Variables and Unary Temporal Logic", *Proceedings, 12th IEEE Symposium on Logic in Computer Science,* 228-235 (1996).

5. U. Hertrampf, C. Lautemann, T. Schwentick, H. Vollmer, K. Wagner, "On the Power of Polynomial-Time Bit Reductions", *Proc. 8th IEEE Conference on Structure in Complexity Theory* (1993) 200-207.
6. N. Immerman, *Descriptive Complexity,* Springer, New York, 1999.
7. N. Immerman and D. Kozen, "Definability with a Bounded Number of Bound Variables", *Information and Computation,* **83**, 121-139 (1989).
8. J. Kamp, *Tense Logic and the Theory of Linear Order,* Ph. D. thesis, UCLA (1968).
9. R. McNaughton and S. Papert, *Counter-Free Automata,* MIT Press, Cambridge, Massachusetts, 1971.
10. S. Margolis, M. Sapir, and P. Weil, "Closed Subgroups in Pro-**V** Topologies and the Extension Problem for Inverse Automata", preprint.
11. J. E. Pin, *Varieties of Formal Languages,* Plenum, London, 1986.
12. J.E. Pin, "$\mathbf{BG} = \mathbf{PG}$: A Success Story" in J. Fountain, ed., *Semigroups, Formal Languages and Groups,* Kluwer Academic Publishers, Dordrecht (1995) 33-48.
13. J.E. Pin, H. Straubing and D. Thérien, "Locally Trivial Categories and Unambiguous Concatenation", *J. Pure and Applied Algebra* **52** (1988) 297-311.
14. J. E. Pin and P. Weil, "Polynomial Closure and Unambiguous Product", *Theory Comput. Systems* **30** (1997) 383-422.
15. J. Rhodes and B. Tilson, "The Kernel of Monoid Morphisms", *J. Pure and Applied Algebra* **62** (1989) 227–268.
16. L. Ribes and P. Zaleskii, "The pro-p topology of a free group and algorithmic problems in semigroups, *International Journal of Algebra and Computation* **4** (1994) 359-374.
17. U. Schöning, "Probablilistic Complexity Classes", *J. Comp. Syst. Sci.* **39** (1989) 84-100.
18. M. P. Schützenberger, "Sur le Produit de Concatenation Non-ambigu", *Semigroup Forum* **13** (1976), 47-76.
19. B. Steinberg, "Finite state automata: a geometric approach, *Trans. Amer. Math. Soc.* **353** (2001) 3409-3464.
20. P. Stiffler, "Extensions of the Fundamental Theorem of Finite Semigroups", *Advances in Mathematics,* **11** 159-209 (1973).
21. H. Straubing, *Finite Automata, Formal Languages, and Circuit Complexity,* Birkhäuser, Boston, 1994.
22. H. Straubing, "When Can One Finite Monoid Simulate Another" in J.C. Birget, S. Margolis, J. Meakin and M. Sapir, eds., *Algorithmic Problems in Groups and Semigroups,* Birkhaüser, Boston (2000) 267-288.
23. H. Straubing, "Families of recognizable sets corresponding to certain varieties of finite monoids", *Journal of Pure and Applied Algebra* **15** (1979), 305-318.
24. H. Straubing, D. Thérien, and W. Thomas, "Regular Languages Defined by Generalized Quantifiers", *Information and Computation* **118** 289-301 (1995).
25. D. Thérien, "Classification of Finite Monoids: the Language Approach" *Theor. Comput. Sci.* **14** (1981) 195-208.
26. D. Thérien and T. Wilke, "Over Words, Two Variables are as Powerful as One Quantifier Alternation," *Proc. 30th ACM Symposium on the Theory of Computing* 256-263 (1998).
27. W. Thomas, "Classifying Regular Events in Symbolic Logic", *J. Computer and System Sciences* **25** (1982) 360-376.
28. B. Tilson, "Categories as Algebra",*J. Pure and Applied Algebra* **48** (1987) 83-198.
29. S. Toda, "PP is as Hard as the Polynomial-Time Hierarchy", *SIAM J. Computing* **20** (1991) 865-877.

30. T. Wilke, "Classifying Discrete Temporal Properties", Habilitationsschrift, University of Kiel, 1998.