

# Semilinearity and Context-Freeness of Languages Accepted by Valence Automata<sup>\*</sup>

P. Buckheister and Georg Zetsche

Fachbereich Informatik, Technische Universität Kaiserslautern

**Abstract.** Valence automata are a generalization of various models of automata with storage. Here, each edge carries, in addition to an input word, an element of a monoid. A computation is considered valid if multiplying the monoid elements on the visited edges yields the identity element. By choosing suitable monoids, a variety of automata models can be obtained as special valence automata. This work is concerned with the accepting power of valence automata. Specifically, we ask for which monoids valence automata can accept only context-free languages or only languages with semilinear Parikh image, respectively. First, we present a characterization of those graph products (of monoids) for which valence automata accept only context-free languages. Second, we provide a necessary and sufficient condition for a graph product of copies of the bicyclic monoid and the integers to yield only languages with semilinear Parikh image when used as a storage mechanism in valence automata. Third, we show that all languages accepted by valence automata over torsion groups have a semilinear Parikh image.

## 1 Introduction

A valence automaton is a finite automaton in which each edge carries, in addition to an input word, an element of a monoid. A computation is considered valid if multiplying the monoid elements on the visited edges yields the identity element. By choosing suitable monoids, one can obtain a wide range of automata with storage mechanisms as special valence automata. Thus, they offer a framework for generalizing insights about automata with storage. For examples of automata as valence automata, see [4,19].

In this work, we are concerned with the accepting power of valence automata. That is, we are interested in relationships between the structure of the monoid representing the storage mechanism and the class of languages accepted by the corresponding valence automata. On the one hand, we address the question for which monoids valence automata accept only *context-free* languages. Since the context-free languages constitute a very well-understood class, insights in this direction promise to shed light on the acceptability of languages by transferring results about context-free languages.

A very well-known result on context-free languages is Parikh's Theorem, which states that the Parikh image (that is, the image under the canonical

---

<sup>\*</sup> The full version of this work is available at <http://arxiv.org/abs/1306.3260>.

morphism onto the free commutative monoid) of each context-free language is semilinear (in this case, the language itself is also called semilinear). It has various applications in proving that certain languages are not context-free and its effective nature (one can actually compute the semilinear representation) allows it to be used in decision procedures for numerous problems (see [15] for an example from group theory and [10] for others). It is therefore our second goal to gain understanding about which monoids cause the corresponding valence automata to accept only languages with a *semilinear Parikh image*.

Our contribution is threefold. First, we obtain a characterization of those graph products (of monoids) whose corresponding valence automata accept only context-free languages. Graph products are a generalization of the free and the direct product in the sense that for each pair of participating factors, it can be specified whether they should commute in the product. Since valence automata over a group accept only context-free languages if and only if the group's word problem (and hence the group itself) can be described by a context-free grammar, such a characterization had already been available for groups in a result by Lohrey and Sénizergues [13]. Therefore, our characterization is in some sense an extension of Lohrey and Sénizergues' to monoids.

Second, we present a necessary and sufficient condition for a graph product of copies of the bicyclic monoid and the integers to yield, when used in valence automata, only languages with semilinear Parikh image. Although this is a smaller class of monoids than arbitrary graph products, it still covers a number of storage mechanisms found in the literature, such as pushdown automata, blind multicounter automata, and partially blind multicounter automata (see [19] for more information). Hence, our result is a generalization of various semilinearity results about these types of automata.

Third, we show that every language accepted by a valence automaton over a torsion group has a semilinear Parikh image. On the one hand, this is particularly interesting because of a result by Render [16], which states that for every monoid  $M$ , the languages accepted by valence automata over  $M$  either (1) coincide with the regular languages, (2) contain the blind one-counter languages, (3) contain the partially blind one-counter languages, or (4) are those accepted by valence automata over an infinite torsion group (which is not locally finite). Hence, our result establishes a strong language theoretic property in the fourth case and thus contributes to completing the picture of language classes that can arise from valence automata.

On the other hand, Lohrey and Steinberg [15] have used the fact that for certain groups, valence automata accept only semilinear languages (in different terms, however) to obtain decidability of the rational subset membership problem. However, their procedures require that the semilinear representation can be obtained effectively. Since there are torsion groups where even the word problem is undecidable [1], our result yields examples of groups that have the semilinearity property but which do not permit the computation of a corresponding representation. Our proof is based on well-quasi-orderings (see, e.g., [11]).

## 2 Basic notions

In this section, we will fix some notation and introduce basic concepts.

A *monoid* is a set  $M$  together with an associative operation and a neutral element. Unless defined otherwise, we will denote the neutral element of a monoid by  $1$  and its operation by juxtaposition. That is, for a monoid  $M$  and elements  $a, b \in M$ ,  $ab \in M$  is their product. In each monoid  $M$ , we have the submonoids  $R(M) = \{a \in M \mid \exists b \in M : ab = 1\}$  and,  $L(M) = \{a \in M \mid \exists b \in M : ba = 1\}$ . When using a monoid  $M$  as part of a control mechanism, the subset  $J(M) = \{a \in M \mid \exists b, c \in M : bac = 1\}$  plays an important role<sup>1</sup>. A *subgroup* of a monoid is a subset that is closed under the operation and is a group.

For an alphabet  $X$ , we will write  $X^*$  for the set of words over  $X$ . The empty word is denoted by  $\lambda \in X^*$ . Given alphabets  $X$  and  $Y$ , subsets of  $X^*$  and  $X^* \times Y^*$  are called *languages* and *transductions*, respectively. A *family* is a set of languages that is closed under isomorphism and contains at least one non-trivial member. For a transduction  $T \subseteq X^* \times Y^*$  and a language  $L \subseteq X^*$ , we write  $T(L) = \{v \in Y^* \mid \exists u \in L : (u, v) \in T\}$ . For any finite subset  $S \subseteq M$  of a monoid, let  $X_S$  be an alphabet in bijection with  $S$ . Let  $\varphi_S : X_S^* \rightarrow M$  be the morphism extending this bijection. Then the set  $\{w \in X_S^* \mid \varphi_S(w) = 1\}$  is called the *identity language of  $M$  with respect to  $S$* .

Let  $\mathcal{F}$  be a family of languages. An  $\mathcal{F}$ -*grammar* is a quadruple  $G = (N, T, P, S)$  where  $N$  and  $T$  are disjoint alphabets and  $S \in N$ .  $P$  is a finite set of pairs  $(A, M)$  with  $A \in N$  and  $M \subseteq (N \cup T)^*$ ,  $M \in \mathcal{F}$ . A pair  $(A, M) \in P$  will also be denoted by  $A \rightarrow M$ . We write  $x \Rightarrow_G y$  if  $x = uAv$  and  $y = uwv$  for some  $u, v, w \in (N \cup T)^*$  and  $(A, M) \in P$  with  $w \in M$ . The *language generated by  $G$*  is  $L(G) = \{w \in T^* \mid S \Rightarrow_G^* w\}$ . Languages generated by  $\mathcal{F}$ -grammars are called *algebraic over  $\mathcal{F}$* . The family of all languages that are algebraic over  $\mathcal{F}$  is called the *algebraic extension* of  $\mathcal{F}$ . The algebraic extension of the family of finite languages is denoted  $\text{CF}$ , its members are called *context-free*.

Given an alphabet  $X$ , we write  $X^\oplus$  for the set of maps  $\alpha : X \rightarrow \mathbb{N}$ . Elements of  $X^\oplus$  are called *multisets*. By way of pointwise addition, written  $\alpha + \beta$ ,  $X^\oplus$  is a monoid. The *Parikh mapping* is the mapping  $\Psi : X^* \rightarrow X^\oplus$  such that  $\Psi(w)(x)$  is the number of occurrences of  $x$  in  $w$  for every  $w \in X^*$  and  $x \in X$ .

Let  $A$  be a (not necessarily finite) set of symbols and  $R \subseteq A^* \times A^*$ . The pair  $(A, R)$  is called a (*monoid*) *presentation*. The smallest congruence of  $A^*$  containing  $R$  is denoted by  $\equiv_R$  and we will write  $[w]_R$  for the congruence class of  $w \in A^*$ . The *monoid presented by  $(A, R)$*  is defined as  $A^*/\equiv_R$ . Note that since we did not impose a finiteness restriction on  $A$ , every monoid has a presentation. By  $\mathbb{B}$ , we denote the monoid presented by  $(A, R)$  with  $A = \{x, \bar{x}\}$  and  $R = (x\bar{x}, \lambda)$  and called *bicyclic monoid*. The elements  $[x]_R$  and  $[\bar{x}]_R$  are called its *positive* and *negative generator*, respectively. The set  $D_1$  of all  $w \in \{x, \bar{x}\}^*$  with  $[w]_R = [\lambda]_R$  is called the *Dyck language*. The group of integers is denoted  $\mathbb{Z}$ . We call  $1 \in \mathbb{Z}$  its *positive* and  $-1 \in \mathbb{Z}$  its *negative generator*.

<sup>1</sup> Note that  $R(M)$ ,  $L(M)$ , and  $J(M)$  are the  $\mathcal{R}$ -,  $\mathcal{L}$ -, and  $\mathcal{J}$ -class, respectively, of the identity and hence are important concepts in the theory of semigroups [7].

Let  $M$  be a monoid. An *automaton over  $M$*  is a tuple  $A = (Q, M, E, q_0, F)$ , in which  $Q$  is a finite set of *states*,  $E$  is a finite subset of  $Q \times M \times Q$  called the set of *edges*,  $q_0 \in Q$  is the *initial state*, and  $F \subseteq Q$  is the set of *final states*. The *step relation*  $\Rightarrow_A$  of  $A$  is a binary relation on  $Q \times M$ , for which  $(p, a) \Rightarrow_A (q, b)$  if and only if there is an edge  $(p, c, q)$  such that  $b = ac$ . The set *generated by  $A$*  is then  $S(A) = \{a \in M \mid \exists q \in F : (q_0, 1) \Rightarrow_A^* (q, a)\}$ . A set  $R \subseteq M$  is called *rational* if it can be written as  $R = S(A)$  for some automaton  $A$  over  $M$ . Rational languages are also called *regular*, the corresponding class is denoted REG. A class  $\mathcal{C}$  for which  $L \in \mathcal{C}$  implies  $T(L) \in \mathcal{C}$  for rational transductions  $T$  is called a *full trio*.

For  $n \in \mathbb{N}$  and  $\alpha \in X^\oplus$ , we use  $n\alpha$  to denote  $\alpha + \dots + \alpha$  ( $n$  summands). A subset  $S \subseteq X^\oplus$  is *linear* if there are elements  $\alpha_0, \dots, \alpha_n$  such that  $S = \{\alpha_0 + \sum_{i=1}^n m_i \alpha_i \mid m_i \in \mathbb{N}, 1 \leq i \leq n\}$ . A set  $S \subseteq C$  is called *semilinear* if it is a finite union of linear sets. In slight abuse of terminology, we will sometimes call a language  $L$  semilinear if the set  $\Psi(L)$  is semilinear.

A *valence automaton over  $M$*  is an automaton  $A$  over  $X^* \times M$ , where  $X$  is an alphabet. Instead of  $A = (Q, X^* \times M, E, q_0, F)$ , we then also write  $A = (Q, X, M, E, q_0, F)$  and for an edge  $(p, (w, m), q) \in E$ , we also write  $(p, w, m, q)$ . The *language accepted by  $A$*  is defined as  $L(A) = \{w \in X^* \mid (w, 1) \in S(A)\}$ . The class of languages accepted by valence automata over  $M$  is denoted by  $\text{VA}(M)$ . It is well-known that  $\text{VA}(M)$  is the smallest full trio containing every identity language of  $M$  (see, for example, [9]).

A *graph* is a pair  $\Gamma = (V, E)$  where  $V$  is a finite set and  $E \subseteq \{S \subseteq V \mid 1 \leq |S| \leq 2\}$ . The elements of  $V$  are called *vertices* and those of  $E$  are called *edges*. If  $\{v\} \in E$  for some  $v \in V$ , then  $v$  is called a *looped vertex*, otherwise it is *unlooped*. A *subgraph* of  $\Gamma$  is a graph  $(V', E')$  with  $V' \subseteq V$  and  $E' \subseteq E$ . Such a subgraph is called *induced (by  $V'$ )* if  $E' = \{S \in E \mid S \subseteq V'\}$ , i.e.  $E'$  contains all edges from  $E$  incident to vertices in  $V'$ . By  $\Gamma \setminus \{v\}$ , for  $v \in V$ , we denote the subgraph of  $\Gamma$  induced by  $V \setminus \{v\}$ . Given a graph  $\Gamma = (V, E)$ , its *underlying loop-free graph* is  $\Gamma' = (V, E')$  with  $E' = E \cap \{S \subseteq V \mid |S| = 2\}$ . For a vertex  $v \in V$ , the elements of  $N(v) = \{w \in V \mid \{v, w\} \in E\}$  are called *neighbors* of  $v$ . Moreover, a *clique* is a graph in which any two distinct vertices are adjacent. A *simple path of length  $n$*  is a sequence  $x_1, \dots, x_n$  of pairwise distinct vertices such that  $\{x_i, x_{i+1}\} \in E$  for  $1 \leq i < n$ . If, in addition, we have  $\{x_n, x_1\} \in E$ , it is called a *cycle*. Such a cycle is called *induced* if  $\{x_i, x_j\} \in E$  implies  $|i - j| = 1$  or  $\{i, j\} = \{1, n\}$ . A loop-free graph  $\Gamma = (V, E)$  is *chordal* if it does not contain an induced cycle of length  $\geq 4$ . It is well-known that every chordal graph contains a vertex whose neighborhood is a clique [3]. By  $C_4$  and  $P_4$ , we denote the cycle of length 4 and the simple path of length 4, respectively. A loop-free graph is called a *transitive forest* if it is the disjoint union of comparability graphs of rooted trees. A result by Wolk [17] states that a loop-free graph is a transitive forest if and only if it contains neither  $C_4$  nor  $P_4$  as an induced subgraph.

Let  $\Gamma = (V, E)$  be a loop-free graph and  $M_v$  a monoid for each  $v \in V$  with a presentation  $(A_v, R_v)$  such that the  $A_v$  are pairwise disjoint. Then the *graph product*  $M = \mathbb{M}(\Gamma, (M_v)_{v \in V})$  is the monoid given by the presentation  $(A, R)$ , where  $A = \bigcup_{v \in V} A_v$  and  $R = \{(ab, ba) \mid a \in A_v, b \in A_w, \{v, w\} \in E\} \cup \bigcup_{v \in V} R_v$ .

Note that for each  $v \in V$ , there is a map  $\varphi_v : M \rightarrow M_v$  such that  $\varphi_v$  is the identity map on  $M_v$ . When  $V = \{0, 1\}$  and  $E = \emptyset$ , we also write  $M_0 * M_1$  for  $M$  and call this the *free product* of  $M_0$  and  $M_1$ . Given a subset  $U \subseteq V$ , we write  $M|_U$  for the product  $\mathbb{M}(\Gamma', (M_v)_{v \in U})$ , where  $\Gamma'$  is the subgraph induced by  $U$ .

Let  $\Gamma = (V, E)$  be a (not necessarily loop-free) graph. Furthermore, for each  $v \in V$ , let  $M_v$  be a copy of  $\mathbb{B}$  if  $v$  is an unlooped vertex and a copy of  $\mathbb{Z}$  if  $v$  is looped. If  $\Gamma^-$  is obtained from  $\Gamma$  by removing all loops, we write  $\mathbb{M}\Gamma$  for the graph product  $\mathbb{M}(\Gamma^-, (M_v)_{v \in V})$ . For information on valence automata over monoids  $\mathbb{M}\Gamma$ , see [19]. For  $i \in \{0, 1\}$ , let  $M_i$  be a monoid and let  $\varphi_i : N \rightarrow M_i$  be an injective morphism. Let  $\equiv$  be the smallest congruence in  $M_0 * M_1$  such that  $\varphi_0(a) \equiv \varphi_1(a)$  for every  $a \in N$ . Then the monoid  $(M_0 * M_1) / \equiv$  is denoted by  $M_0 *_N M_1$  and called a *free product with amalgamation*.

### 3 Auxiliary Results

In this section, we present auxiliary results that are used in later sections. In the following, we will call a monoid  $M$  an *FRI-monoid* (or say that  $M$  has the FRI-property) if for every finitely generated submonoid  $N$  of  $M$ , the set  $R(N)$  is finite. In [16] and independently in [18], the following was shown.

**Theorem 1.**  $\text{VA}(M) = \text{REG}$  if and only if  $M$  is an FRI-monoid.

The first lemma states a well-known fact from semigroup theory.

**Lemma 1.** For each monoid  $M$ , exactly one of the following holds: Either  $J(M)$  is a group or  $M$  contains a copy of  $\mathbb{B}$  as a submonoid.

We will employ a result by van Leeuwen [12] stating that semilinearity of all languages is preserved by building the algebraic extension of a language family.

**Theorem 2.** Let  $\mathcal{F}$  be a family of semilinear languages. Then every language that is algebraic over  $\mathcal{F}$  is also semilinear.

In light of the previous theorem, the following implies that the class of monoids  $M$  for which  $\text{VA}(M)$  contains only semilinear languages is closed under taking free products with amalgamation over a finite identified subgroup that contains the identity of each factor. In the case where the factors are residually finite groups, this was already shown in [15, Lemma 8] (however, for a more general operation than free products with amalgamation). The following also implies that if  $\text{VA}(M_i)$  contains only context-free languages for  $i \in \{0, 1\}$ , then this is also true for  $\text{VA}(M_0 *_F M_1)$ .

**Theorem 3.** For each  $i \in \{0, 1\}$ , let  $M_i$  be a finitely generated monoid and  $F$  be a subgroup that contains  $M_i$ 's identity. Every language in  $\text{VA}(M_0 *_F M_1)$  is algebraic over  $\text{VA}(M_0) \cup \text{VA}(M_1)$ .

*Proof.* Since the algebraic extension of a full trio is again a full trio, it suffices to show that with respect to some generating set  $S \subseteq M_0 *_F M_1$ , the identity language of  $M_0 *_F M_1$  is algebraic over  $\text{VA}(M_0) \cup \text{VA}(M_1)$ .

For  $i \in \{0, 1\}$ , let  $S_i \subseteq M_i$  be a finite generating set for  $M_i$  such that  $F \subseteq S_i$ . Furthermore, let  $X_i$  be an alphabet in bijection with  $S_i$  and let  $\varphi_i : X_i^* \rightarrow M_i$  be the morphism extending this bijection. Moreover, let  $Y_i \subseteq X_i$  be the subset with  $\varphi_i(Y_i) = F$ . Let  $\psi_i : M_i \rightarrow M_0 *_F M_1$  be the canonical morphism. Since  $F$  is a subgroup of  $M_0$  and  $M_1$ ,  $\psi_0$  and  $\psi_1$  are injective (see e.g. [7, Theorem 8.6.1]). Let  $X = X_0 \cup X_1$  and let  $\varphi : X^* \rightarrow M_0 *_F M_1$  be the morphism extending  $\psi_0\varphi_0$  and  $\psi_1\varphi_1$ . Then the identity language of  $M_0 *_F M_1$  is  $\varphi^{-1}(1)$  and we shall prove the theorem by showing that  $\varphi^{-1}(1)$  is algebraic over  $\text{VA}(M_0) \cup \text{VA}(M_1)$ . We will make use of the following fact about free products with amalgamation of monoids with a finite identified subgroup. Let  $s_1, \dots, s_n, s'_1, \dots, s'_m \in (X_0^* \setminus \varphi_0^{-1}(F)) \cup (X_1^* \setminus \varphi_1^{-1}(F))$ , such that  $s_j \in X_i^*$  if and only if  $s_{j+1} \in X_{1-i}^*$  for  $1 \leq j < n$ ,  $i \in \{0, 1\}$  and  $s'_j \in X_i^*$  if and only if  $s'_{j+1} \in X_{1-i}^*$  for  $1 \leq j < m$ ,  $i \in \{0, 1\}$ . Then the equality  $\varphi(s_1 \cdots s_n) = \varphi(s'_1 \cdots s'_m)$  implies  $n = m$ . A stronger statement was shown in [14, Lemma 10]. We will refer to this as the *syllable property*.

For each  $i \in \{0, 1\}$  and  $f \in F$ , we define  $L_{i,f} = \varphi_i^{-1}(f)$  and write  $y_f$  for the symbol in  $Y_i$  with  $\varphi_i(y_f) = f$ . Then clearly  $L_{i,1} \in \text{VA}(M_i)$ . Furthermore, since  $L_{i,f} = \{w \in X_i^* \mid y_f w \in L_{i,1}\}$ , (here we again use that  $F$  is a group) we can obtain  $L_{i,f}$  from  $L_{i,1}$  by a rational transduction and hence  $L_{i,f} \in \text{VA}(M_i)$ .

Let  $\mathcal{F} = \text{VA}(M_0) \cup \text{VA}(M_1)$ . Since for each  $\mathcal{F}$ -grammar  $G$ , it is clearly possible to construct an  $\mathcal{F}$ -grammar  $G'$  such that  $L(G')$  consists of all sentential forms of  $G$ , it suffices to construct an  $\mathcal{F}$ -grammar  $G = (N, T, P, S)$  with  $N \cup T = X$  and  $S \Rightarrow_G^* w$  if and only if  $\varphi(w) = 1$  for  $w \in X^*$ . We construct  $G = (N, T, P, S)$  as follows. Let  $N = Y_0 \cup Y_1$  and  $T = (X_0 \cup X_1) \setminus (Y_0 \cup Y_1)$ . As productions, we have  $y \rightarrow L_{1-i,f}$  for each  $y \in Y_i$  where  $f = \varphi_i(y)$ . Since  $1 \in F$ , we have an  $e_i \in Y_i$  with  $\varphi_i(e_i) = 1$ . As the start symbol, we choose  $S = e_0$ . We claim that for  $w \in X^*$ , we have  $S \Rightarrow_G^* w$  if and only if  $\varphi(w) = 1$ .

The “only if” is clear. Let  $w \in X^*$  with  $\varphi(w) = 1$ . Write  $w = w_1 \cdots w_n$  such that  $w_j \in X_0^* \cup X_1^*$  for all  $1 \leq j \leq n$  such that  $w_j \in X_i^*$  if and only if  $w_{j+1} \in X_{1-i}^*$  for  $i \in \{0, 1\}$  and  $1 \leq j < n$ . We show by induction on  $n$  that  $S \Rightarrow_G^* w$ . For  $n \leq 1$ , we have  $w \in X_i^*$  for some  $i \in \{0, 1\}$ . Since  $1 = \varphi(w) = \psi_i(\varphi_i(w))$  and  $\psi_i$  is injective, we have  $\varphi_i(w) = 1$  and hence  $w \in L_{i,1}$ . This means  $S = e_0 \Rightarrow_G w$  or  $S = e_0 \Rightarrow_G e_1 \Rightarrow_G w$ , depending on whether  $i = 1$  or  $i = 0$ .

Now let  $n \geq 2$ . We claim that there is a  $1 \leq j \leq n$  with  $\varphi(w_j) \in F$ . Indeed, if  $\varphi(w_j) \notin F$  for all  $1 \leq j \leq n$  and since  $\varphi(w_1 \cdots w_n) = 1 = \varphi(\lambda)$ , the syllable property implies  $n = 0$ , against our assumption. Hence, let  $f = \varphi(w_j) \in F$ . Furthermore, let  $w_j \in X_i^*$  and choose  $y \in Y_{1-i}$  so that  $\varphi_{1-i}(y) = f$ . Then  $\psi_i(\varphi_i(w_j)) = \varphi(w_j) = f$  and the injectivity of  $\psi_i$  yields  $\varphi_i(w_j) = f$ . Hence,  $w_j \in L_{i,f}$  and thus  $w' = w_1 \cdots w_{j-1} y w_{j+1} \cdots w_n \Rightarrow_G w$ . For  $w'$  the induction hypothesis holds, meaning  $S \Rightarrow_G^* w'$  and thus  $S \Rightarrow_G^* w$ .  $\square$

## 4 Context-Freeness

In this section, we are concerned with the context-freeness of languages accepted by valence automata over graph products. The first lemma is a simple observation and we will not provide a proof. In the case of groups, it appeared in [5].

**Lemma 2.** *Let  $\Gamma = (V, E)$  and  $M = \mathbb{M}(\Gamma, (M_v)_{v \in V})$  be a graph product. Then for each  $v \in V$ , we have  $M \cong (M \upharpoonright_{V \setminus \{v\}}) *_{M \upharpoonright_{N(v)}} (M \upharpoonright_{N(v)} \times M_v)$ .*

The following is a result by Lohrey and S enizergues [13].

**Theorem 4.** *Let  $G_v$  be a non-trivial group for each  $v \in V$ . Then  $\mathbb{M}(\Gamma, (G_v)_{v \in V})$  is virtually free if and only if (1) for each  $v \in V$ ,  $G_v$  is virtually free, (2) if  $G_v$  and  $G_w$  are infinite and  $v \neq w$ , then  $\{v, w\} \notin E$ , (3) if  $G_v$  is infinite,  $G_u$  and  $G_w$  are finite and  $\{v, u\}, \{v, w\} \in E$ , then  $\{u, w\} \in E$ , and (4)  $\Gamma$  is chordal.*

Aside from Theorem 3, the following is the key tool to prove our result on context-freeness. We call a monoid  $M$  *context-free* if  $\text{VA}(M) \subseteq \text{CF}$ .

**Lemma 3.** *The direct product of monoids  $M_0$  and  $M_1$  is context-free if and only if for some  $i \in \{0, 1\}$ ,  $M_i$  is context-free and  $M_{1-i}$  is an FRI-monoid.*

*Proof.* Suppose  $M_i$  is context-free and  $M_{1-i}$  is an FRI-monoid. Then each language  $L \in \text{VA}(M_i \times M_{1-i})$  is contained in  $\text{VA}(M_i \times N)$  for some finitely generated submonoid  $N$  of  $M_{1-i}$ . Since  $M_{1-i}$  is an FRI-monoid,  $N$  has finitely many right-invertible elements and hence  $J(N)$  is a finite group. Since no element outside of  $J(N)$  can appear in a product yielding the identity, we may assume that  $L \in \text{VA}(M_i \times J(N))$ . This means, however, that  $L$  can be accepted by a valence automaton over  $M_i$  by keeping the right component of the storage monoid in the state of the automaton. Hence,  $L \in \text{VA}(M_i)$  is context-free.

Suppose  $\text{VA}(M_0 \times M_1) \subseteq \text{CF}$ . Then certainly  $\text{VA}(M_i) \subseteq \text{CF}$  for each  $i \in \{0, 1\}$ . This means we have to show that at least one of the monoids  $M_0$  and  $M_1$  is an FRI-monoid and thus, toward a contradiction, assume that none of them is.<sup>2</sup>

By Lemma 1, for each  $i$ , either  $J(M_i)$  is a subgroup of  $M_i$  or  $M_i$  contains a copy of  $\mathbb{B}$  as a submonoid. Since every infinite virtually free group contains an element of infinite order, we have that for each  $i$ , either (1)  $J(M_i)$  is an infinite group and hence contains a copy of  $\mathbb{Z}$  or (2)  $M_i$  contains a copy of  $\mathbb{B}$ . In any case,  $\text{VA}(M_0 \times M_1)$  contains  $\{a^n b^m c^n d^m \mid n, m \geq 0\}$ , which is not context-free.  $\square$

We are now ready to prove our main result on context-freeness. Since for a graph product  $M = \mathbb{M}(\Gamma, (M_v)_{v \in V})$ , there is a morphism  $\varphi_v : M \rightarrow M_v$  for each  $v \in V$  that restricts to the identity on  $M_v$ , we have  $J(M) \cap M_v = J(M_v)$ : While the inclusion “ $\supseteq$ ” is true for any submonoid, given  $b \in J(M) \cap M_v$  with  $abc = 1$ ,  $a, c \in M$ , we also have  $\varphi_v(a)b\varphi_v(c) = \varphi_v(abc) = 1$  and hence  $b \in J(M_v)$ . This means no element of  $M_v \setminus J(M_v)$  can appear in a product yielding the identity. In particular, removing a vertex  $v$  with  $J(M_v) = \{1\}$  will not change  $\text{VA}(M)$ . Hence, our requirement that  $J(M_v) \neq \{1\}$  is not a serious restriction.

**Theorem 5.** *Let  $\Gamma = (V, E)$  and let  $J(M_v) \neq \{1\}$  for any  $v \in V$ .  $M = \mathbb{M}(\Gamma, (M_v)_{v \in V})$  is context-free if and only if (1) for each  $v \in V$ ,  $M_v$  is context-free,*

<sup>2</sup> In the full version, we have a second proof for the fact that  $\text{VA}(M_0 \times M_1)$  contains non-context-free languages in this case. It is elementary in the sense that it does not invoke the fact that context-free groups are virtually free.

- (2) if  $M_v$  and  $M_w$  are not FRI-monoids and  $v \neq w$ , then  $\{v, w\} \notin E$ ,
- (3) if  $M_v$  is not an FRI-monoid,  $M_u$  and  $M_w$  are FRI-monoids and  $\{v, u\}, \{v, w\} \in E$ , then  $\{u, w\} \in E$ , and
- (4) the graph  $\Gamma$  is chordal.

*Proof.* First, we show that conditions (1)–(4) are necessary. For (1), this is immediate and for (2), this follows from Lemma 3. If (3) is violated then for some  $u, v, w \in V$ ,  $M_v \times (M_u * M_w)$  is a submonoid of  $M$  such that  $M_u$  and  $M_w$  are FRI-monoids and  $M_v$  is not. Since  $M_u$  and  $M_w$  contain non-trivial (finite) subgroups,  $M_u * M_w$  contains an infinite group and is thus not an FRI-monoid, meaning  $M_v \times (M_u * M_w)$  is not context-free by Lemma 3.

Suppose (4) is violated for context-free  $M$ . By (2) and (3), any induced cycle of length at least four involves only vertices with FRI-monoids. Each of these, however, contains a non-trivial finite subgroup. This means  $M$  contains an induced cycle graph product of non-trivial finite groups, which is not virtually free by Theorem 4 and hence has a non-context-free identity language.

In order to prove the other direction, we note that  $\text{VA}(M) \subseteq \text{CF}$  follows if  $\text{VA}(M') \subseteq \text{CF}$  for every finitely generated submonoid  $M' \subseteq M$ . Since every such submonoid is contained in a graph product  $N = \mathbb{M}(\Gamma, (N_v)_{v \in V})$  where each  $N_v$  is a finitely generated submonoid of  $M_v$ , it suffices to show that for such graph products, we have  $\text{VA}(N) \subseteq \text{CF}$ . This means whenever  $M_v$  is an FRI-monoid,  $N_v$  has finitely many right-invertible elements. Moreover, since  $N_v \cap \text{J}(N) = \text{J}(N_v)$ , no element of  $N_v \setminus \text{J}(N_v)$  can appear in a product yielding the identity. Hence, if  $N_v$  is generated by  $S \subseteq N_v$ , replacing  $N_v$  by the submonoid generated by  $S \cap \text{J}(N_v)$  does not change the identity languages of the graph product. Thus, we assume that each  $N_v$  is generated by a finite subset of  $\text{J}(N_v)$ . Therefore, whenever  $M_v$  is an FRI-monoid,  $N_v$  is a finite group.

We first establish sufficiency in the case that  $M_v$  is an FRI-monoid for every  $v \in V$  and proceed by induction on  $|V|$ . This means that  $N_v$  is a finite group for every  $v \in V$ . Since  $\Gamma$  is chordal, there is a  $v \in V$  whose neighborhood is a clique. This means  $N \upharpoonright_{N(v)}$  is a finite group and hence  $N \upharpoonright_{N(v)} \times N_v$  context-free by Lemma 3. Since  $N \upharpoonright_{V \setminus \{v\}}$  is context-free by induction, Theorem 3 and Lemma 2 imply that  $N$  is context-free.

To complete the proof, suppose there are  $n$  vertices  $v \in V$  for which  $M_v$  is not an FRI-monoid. We proceed by induction on  $n$ . The case  $n = 0$  is treated above. Choose  $v \in V$  such that  $M_v$  is not an FRI-monoid. For each  $u \in N(v)$ ,  $M_u$  is an FRI-monoid by condition (2), and hence  $N_u$  a finite group. Furthermore, condition (3) guarantees that  $N(v)$  is a clique and hence  $N \upharpoonright_{N(v)}$  is a finite group. As above, Theorem 3 and Lemma 2 imply that  $N$  is context-free.  $\square$

## 5 Semilinearity

A well-known theorem by Chomsky and Schützenberger [2] was re-proved and phrased in terms of valence automata in the following way by Kambites [9].

**Theorem 6.**  $\text{VA}(\mathbb{Z} * \mathbb{Z}) = \text{CF}$ .



Using standard methods of formal language theory, one can show:

**Lemma 4 ([15,19]).** *If every language in  $\text{VA}(M)$  is semilinear, then so is every language in  $\text{VA}(M \times \mathbb{Z})$ .*

The following is a consequence of the results of Greibach [6] and Jantzen [8].

**Lemma 5.**  $\text{VA}(\mathbb{B} \times \mathbb{B})$  *contains a non-semilinear language.*

The next result also appears in [19], where, however, it was not made explicit that the language is unary. A proof can be found in the full version.

**Lemma 6.** *If  $\Gamma$ 's underlying loop-free graph contains  $P_4$  as an induced subgraph, then  $\text{VA}(\mathbb{M}\Gamma)$  contains an undecidable unary language.*

We are now ready to show the first main result of this section. Note that the first condition of the following theorem is similar to conditions (2) and (3) in Theorem 5 (and 4): we have looped vertices instead of FRI-monoids (finite groups) and unlooped vertices instead of non-FRI-monoids (infinite groups).

**Theorem 7.** *All languages in  $\text{VA}(\mathbb{M}\Gamma)$  are semilinear if and only if (1)  $\Gamma$  contains neither  $\bullet \rightarrow \bullet$  nor  $\circ \rightarrow \bullet \rightarrow \circ$  as an induced subgraph and (2)  $\Gamma$ 's underlying loop-free graph contains neither  $C_4$  nor  $P_4$  as an induced subgraph.*

*Proof.* First, observe that if  $\text{VA}(N_i) \subseteq \text{VA}(M_i)$  for  $i = 0, 1$  then  $\text{VA}(N_0 \times N_1) \subseteq \text{VA}(M_0 \times M_1)$ . Let  $\Gamma = (V, E)$ . Suppose conditions 1 and 2 hold. We proceed by induction on  $|V|$ . 2 implies that  $\Gamma$ 's underlying loop-free graph is a transitive forest. If  $\Gamma$  is not connected, then  $\mathbb{M}\Gamma$  is a free product of graph products  $\mathbb{M}\Gamma_1$  and  $\mathbb{M}\Gamma_2$ , for which  $\text{VA}(\mathbb{M}\Gamma_i)$  contains only semilinear languages by induction. Hence, by Theorems 2 and 3, every language in  $\text{VA}(\mathbb{M}\Gamma)$  is semilinear. If  $\Gamma$  is connected, there is a vertex  $v \in V$  that is adjacent to every vertex other than itself. We distinguish two cases.

If  $v$  is a looped vertex, then  $\text{VA}(\mathbb{M}\Gamma) = \text{VA}(\mathbb{Z} \times \mathbb{M}(\Gamma \setminus \{v\}))$ , which contains only semilinear languages by induction and Lemma 4. If  $v$  is an unlooped vertex, then by 1,  $V \setminus \{v\}$  induces a clique of looped vertices. Thus,  $\mathbb{M}\Gamma \cong \mathbb{B} \times \mathbb{Z}^{|V|-1}$ , meaning  $\text{VA}(\mathbb{M}\Gamma)$  contains only semilinear languages by Lemma 4.

We shall now prove the other direction. If  $\Gamma$  contains  $\bullet \rightarrow \bullet$  as an induced subgraph, then  $\text{VA}(\mathbb{B} \times \mathbb{B})$  is included in  $\text{VA}(\mathbb{M}\Gamma)$  and the former contains a non-semilinear language by Lemma 5. If  $\Gamma$  contains  $\circ \rightarrow \bullet \rightarrow \circ$ , then  $\mathbb{M}\Gamma$  contains a copy of  $\mathbb{B} \times (\mathbb{Z} * \mathbb{Z})$  as a submonoid. By Theorem 6, we have  $\text{VA}(\mathbb{B}) \subseteq \text{VA}(\mathbb{Z} * \mathbb{Z})$  and hence the observation above implies  $\text{VA}(\mathbb{B} \times \mathbb{B}) \subseteq \text{VA}(\mathbb{B} \times (\mathbb{Z} * \mathbb{Z}))$ .

Suppose  $\Gamma$ 's underlying loop-free graph contains  $C_4$  as an induced subgraph. Since we have already shown that the presence of  $\bullet \rightarrow \bullet$  or  $\circ \rightarrow \bullet \rightarrow \circ$  as an induced subgraph guarantees a non-semilinear language in  $\text{VA}(\mathbb{M}\Gamma)$ , we may assume that all four participating vertices are looped. Hence,  $\mathbb{M}\Gamma$  contains a copy of  $(\mathbb{Z} * \mathbb{Z}) \times (\mathbb{Z} * \mathbb{Z})$ . By Theorem 6 and the observation above, this means  $\text{VA}(\mathbb{B} \times \mathbb{B}) \subseteq \text{VA}(\mathbb{M}\Gamma)$ . Thus,  $\text{VA}(\mathbb{M}\Gamma)$  contains non-semilinear languages. Finally, if  $\Gamma$ 's underlying loop-free graph contains  $P_4$  as an induced subgraph, Lemma 6 provides the existence of an undecidable unary language in  $\text{VA}(\mathbb{M}\Gamma)$ . Since such a language cannot be semilinear, the lemma is proven.  $\square$

**Torsion groups** A *torsion group* is a group  $G$  in which for each  $g \in G$ , there is a  $k \in \mathbb{N} \setminus \{0\}$  with  $g^k = 1$ . In the following, we show that for torsion groups  $G$ , all languages in  $\text{VA}(G)$  are semilinear. The key ingredient in our proof is showing that a certain set of multisets is upward closed with respect to a well-quasi-ordering. A *well-quasi-ordering on  $A$*  is a reflexive transitive relation  $\leq$  on  $A$  such that for every infinite sequence  $(a_n)_{n \in \mathbb{N}}$ ,  $a_n \in A$ , there are indices  $i < j$  with  $a_i \leq a_j$ . We call a subset  $B \subseteq A$  *upward closed* if  $a \in B$  and  $a \leq b$  imply  $b \in B$ . A basic observation about well-quasi-ordered sets states that for each upward closed set  $B \subseteq A$ , the set of its minimal elements is finite and  $B$  is the set of those  $a \in A$  with  $m \leq a$  for some minimal  $m \in B$  (see [11]).

Given multisets  $\alpha, \beta \in X^\oplus$  and  $k \in \mathbb{N}$ , we write  $\alpha \equiv_k \beta$  if  $\alpha(x) \equiv \beta(x) \pmod{k}$  for each  $x \in X$ . We write  $\alpha \leq_k \beta$  if  $\alpha \leq \beta$  and  $\alpha \equiv_k \beta$ . Clearly,  $\leq_k$  is a well-quasi-ordering on  $X^\oplus$ : Since  $\equiv_k$  has finite index in  $X^\oplus$ , we find in any infinite sequence  $\alpha_1, \alpha_2, \dots \in X^\oplus$  an infinite subsequence  $\alpha'_1, \alpha'_2, \dots \in X^\oplus$  of  $\equiv_k$ -equivalent multisets. Furthermore,  $\leq$  is well-known to be a well-quasi-ordering and yields indices  $i < j$  with  $\alpha'_i \leq \alpha'_j$  and hence  $\alpha'_i \leq_k \alpha'_j$ . If  $S \subseteq X^\oplus$  is upward closed with respect to  $\leq_k$ , we also say  $S$  is  *$k$ -upward-closed*. The observation above means in particular that every  $k$ -upward-closed set is semilinear.

**Theorem 8.** *For every torsion group  $G$ , the languages in  $\text{VA}(G)$  are semilinear.*

*Proof.* Let  $G$  be a torsion group and  $K$  be accepted by the valence automaton  $A = (Q, X, G, E, q_0, F)$ . We regard the finite set  $E$  as an alphabet and define the automaton  $\hat{A} = (Q, E, G, \hat{E}, q_0, F)$  such that  $\hat{E} = \{(p, (p, w, g, q), g, q) \mid (p, w, g, q) \in E\}$ . Let  $\hat{K} = L(\hat{A})$ . Clearly, in order to prove Theorem 8, it suffices to show that  $\hat{K}$  is semilinear.

For a word  $w \in E^*$ ,  $w = (p_1, x_1, g_1, q_1) \cdots (p_n, x_n, g_n, q_n)$ , we write  $\sigma(w)$  for the set  $\{p_i, q_i \mid 1 \leq i \leq n\}$ .  $w$  is called a  *$p, q$ -computation* if  $p_1 = p$ ,  $q_n = q$ , and  $q_i = p_{i+1}$  for  $1 \leq i < n$ . A  *$q, q$ -computation* is also called a  *$q$ -loop*. Moreover, a  $q$ -loop  $w$  is called *simple* if  $q_i \neq q_j$  for  $i \neq j$ .

For each  $S \subseteq Q$ , let  $F_S$  be the set of all words  $w \in E^*$  with  $\sigma(w) = S$  and for which there is a  $q \in F$  such that  $w$  is a  $q_0, q$ -computation and  $|w| \leq |Q| \cdot (2^{|Q|} + 1)$ . Let  $L_S \subseteq E^*$  consist of all  $w \in E^*$  such that  $w$  is a simple  $q$ -loop for some  $q \in S$  and  $\sigma(w) \subseteq S$ . Note that  $L_S$  is finite, which allows us to define the alphabet  $Y_S$  so as to be in bijection with  $L_S$ . Let  $\varphi : Y_S \rightarrow L_S$  be this bijection and let  $\tilde{\varphi} : Y_S^\oplus \rightarrow E^\oplus$  be the morphism with  $\tilde{\varphi}(y) = \Psi(\varphi(y))$  for  $y \in Y_S$ .

For  $p, q$ -computations  $v, w \in E^*$ , we write  $v \vdash w$  if  $\sigma(v) = \sigma(w)$  and  $w = rst$  such that  $r$  is a  $p, q'$ -computation,  $s$  is a simple  $q'$ -loop,  $t$  is a  $q', q$ -computation, and  $v = rt$ . Moreover, let  $\preceq$  be the reflexive transitive closure of  $\vdash$ . In other words,  $v \preceq w$  means that  $w$  can be obtained from  $v$  by inserting simple  $q$ -loops for states  $q \in Q$  without increasing the set of visited states. For each  $v \in F_S$ , we define

$$U_v = \{\mu \in Y_S^\oplus \mid \exists w \in \hat{K} : v \preceq w, \Psi(w) = \Psi(v) + \tilde{\varphi}(\mu)\}$$

(note that there is only one  $S \subseteq Q$  with  $v \in F_S$ ). We claim that

$$\Psi(\hat{K}) = \bigcup_{S \subseteq Q} \bigcup_{v \in F_S} \Psi(v) + \tilde{\varphi}(U_v). \quad (*)$$

The inclusion “ $\supseteq$ ” holds by definition. For the other direction, we show by induction on  $n$  that for any  $q_f \in F$  and any  $q_0, q_f$ -computation  $w \in E^*$ ,  $|w| = n$ , there is a  $v \in F_S$  for  $S = \sigma(w)$  and a  $\mu \in Y_S^\oplus$  with  $v \preceq w$  and  $\Psi(w) = \Psi(v) + \tilde{\varphi}(\mu)$ . If  $|w| \leq |Q| \cdot (2^{|Q|} + 1)$ , this is satisfied by  $v = w$  and  $\mu = 0$ . Therefore, assume  $|w| > |Q| \cdot (2^{|Q|} + 1)$  and write  $w = (p_1, x_1, g_1, q_1) \cdots (p_n, x_n, g_n, q_n)$ . Since  $n = |w| > |Q| \cdot (2^{|Q|} + 1)$ , there is a  $q \in Q$  that appears more than  $2^{|Q|} + 1$  times in the sequence  $q_1, \dots, q_n$ . Therefore, we can write

$$w = w_0(p'_1, x'_1, g'_1, q)w_1 \cdots (p'_m, x'_m, g'_m, q)w_m$$

with  $m > 2^{|Q|} + 1$ . Observe that for each  $1 \leq i < m$ , the word  $w_i(p'_{i+1}, x'_{i+1}, g'_{i+1}, q)$  is a  $q$ -loop. Since  $m - 1 > 2^{|Q|}$ , there are indices  $1 \leq i < j < m$  with  $\sigma(w_i(p'_{i+1}, x'_{i+1}, g'_{i+1}, q)) = \sigma(w_j(p'_{j+1}, x'_{j+1}, g'_{j+1}, q))$ . Furthermore, we can find a simple  $q$ -loop  $\ell$  as a subword of  $w_i(p'_{i+1}, x'_{i+1}, g'_{i+1}, q)$ . This means for the word  $w' \in E^*$ , which is obtained from  $w$  by removing  $\ell$ , we have  $\sigma(w') = \sigma(w)$  and thus  $w' \vdash w$ . Moreover, with  $S = \sigma(w)$  and  $\varphi(y) = \ell$ ,  $y \in Y_S$ , we have  $\Psi(w) = \Psi(w') + \tilde{\varphi}(y)$ . Finally, since  $|w'| < |w|$ , the induction hypothesis guarantees a  $v \in F_S$  and a  $\mu \in Y_S^\oplus$  with  $v \preceq w'$  and  $\Psi(w') = \Psi(v) + \tilde{\varphi}(\mu)$ . We have  $v \preceq w$  and  $\Psi(w) = \Psi(v) + \tilde{\varphi}(\mu + y)$  and the induction is complete. In order to prove “ $\subseteq$ ” of (\*), suppose  $w \in \hat{K}$ . Since  $w$  is a  $q_0, q_f$ -computation for some  $q_f \in F$ , we can find the above  $v \in F_S$ ,  $S = \sigma(w)$ , and  $\mu \in Y_S^\oplus$  with  $v \preceq w$  and  $\Psi(w) = \Psi(v) + \tilde{\varphi}(\mu)$ . This means  $\mu \in U_v$  and hence  $\Psi(w)$  is contained in the right hand side of (\*). This proves (\*).

By (\*) and since  $F_S$  is finite for each  $S \subseteq Q$ , it suffices to show that  $U_v$  is semilinear for each  $v \in F_S$  and  $S \subseteq Q$ . Let  $\gamma : E^* \rightarrow G$  be the morphism with  $\gamma((p, x, g, q)) = g$  for  $(p, x, g, q) \in E$ . Since  $G$  is a torsion group, the finiteness of  $L_S$  permits us to choose a  $k \in \mathbb{N}$  such that  $\gamma(\ell)^k = 1$  for any  $\ell \in L_S$ . We claim that  $U_v$  is  $k$ -upward-closed. It suffices to show that for  $\mu \in U_v$ , we also have  $\mu + k \cdot y \in U_v$  for any  $y \in Y_S$ . Hence, let  $\mu \in U_v$  with  $w \in \hat{K}$  such that  $v \preceq w$  and  $\Psi(w) = \Psi(v) + \tilde{\varphi}(\mu)$  and let  $\mu' = \mu + k \cdot y$ . Let  $\ell = \varphi(y) \in L_S$  be a simple  $q$ -loop. Then  $q \in S$  and since  $\sigma(w) = \sigma(v) = S$ , we can write  $w = r(q_1, x_1, g_1, q)s$ ,  $r, s \in E^*$ . The fact that  $w \in \hat{K}$  means in particular  $\gamma(w) = 1$ . Thus, the word  $w' = r(q_1, x_1, g_1, q)\ell^k s$  is a  $q_0, q_f$ -computation for some  $q_f \in F$  with  $\gamma(w') = 1$  since  $\gamma(\ell)^k = 1$ . This means  $w' \in \hat{K}$  and  $\Psi(w') = \Psi(w) + k \cdot \Psi(\ell) = \Psi(v) + \tilde{\varphi}(\mu + k \cdot y)$ . We also have  $\sigma(\ell) \subseteq S$  and hence  $v \preceq w \preceq w'$ . Thus,  $\mu' = \mu + k \cdot y \in U_v$ . This proves  $U_v$  to be  $k$ -upward-closed and thus semilinear.  $\square$

Render [16] proved that for every monoid  $M$ , the class  $\text{VA}(M)$  either (1) coincides with the regular languages, (2) contains the blind one-counter languages, (3) contains the partially blind one-counter languages, or (4) consists of those accepted by valence automata over an infinite torsion group. Hence, we obtain:

**Corollary 1.** *For each monoid  $M$ , at least one of the following holds:*

- (1)  $\text{VA}(M)$  contains only semilinear languages.
- (2)  $\text{VA}(M)$  contains the languages of blind one-counter automata.
- (3)  $\text{VA}(M)$  contains the languages of partially blind one-counter automata.

There are torsion groups with an undecidable word problem [1], hence:

**Corollary 2.** *There is a group  $G$  with an undecidable word problem such that all languages in  $\text{VA}(G)$  are semilinear.*

As another application, we can show that the one-sided Dyck language is not accepted by any valence automaton over  $G \times \mathbb{Z}^n$  where  $G$  is a torsion group.

**Corollary 3.** *For torsion groups  $G$  and  $n \in \mathbb{N}$ , we have  $D_1 \notin \text{VA}(G \times \mathbb{Z}^n)$ .*

*Acknowledgements* We are indebted to one of the anonymous referees, who pointed out a misuse of terminology in a previous version of Theorem 3.

## References

1. Adian, S.I.: The Burnside problem and related topics. *Russian Mathematical Surveys* 65(5), 805–855 (2010)
2. Chomsky, N., Schützenberger, M.P.: The algebraic theory of context-free languages. In: *Computer Programming and Formal Systems*, pp. 118–161. North-Holland, Amsterdam (1963)
3. Dirac, G.: On rigid circuit graphs. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* 25(1-2), 71–76 (1961)
4. Gilman, R.H.: Formal Languages and Infinite Groups, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 25, pp. 27–51. American Mathematical Society (1996)
5. Green, R.E.: Graph Products of Groups. Ph.D. thesis, University of Leeds (1990)
6. Greibach, S.A.: Remarks on blind and partially blind one-way multicounter machines. *Theoretical Computer Science* 7(3), 311–324 (1978)
7. Howie, J.M.: *Fundamentals of Semigroup Theory*. Clarendon Press, Oxford (1995)
8. Jantzen, M.: Eigenschaften von Petrinetzsprachen. Ph.D. thesis, Universität Hamburg (1979)
9. Kambites, M.: Formal languages and groups as memory. *Communications in Algebra* 37, 193–208 (2009)
10. Kopczynski, E., To, A.: Parikh images of grammars: Complexity and applications. In: *Proceedings of LICS 2010*. pp. 80–89 (2010)
11. Kruskal, J.B.: The theory of well-quasi-ordering: A frequently discovered concept. *Journal of Combinatorial Theory, Series A* 13(3), 297–305 (1972)
12. van Leeuwen, J.: A generalisation of Parikh’s theorem in formal language theory. In: *ICALP 1974, LNCS*, vol. 14, pp. 17–26. Springer (1974)
13. Lohrey, M., Sénizergues, G.: When is a graph product of groups virtually-free? *Communications in Algebra* 35(2), 617–621 (2007)
14. Lohrey, M., Sénizergues, G.: Rational subsets in HNN-extensions and amalgamated products. *Internat. J. Algebra Comput.* 18(01), 111–163 (2008)
15. Lohrey, M., Steinberg, B.: The submonoid and rational subset membership problems for graph groups. *J. Algebra* 320(2), 728–755 (2008)
16. Render, E.: Rational Monoid and Semigroup Automata. Ph.D. thesis, University of Manchester (2010)
17. Wolk, E.S.: A note on “the comparability graph of a tree”. *Proceedings of the American Mathematical Society* 16(1), 17–20 (1965)
18. Zetsche, G.: On the capabilities of grammars, automata, and transducers controlled by monoids. In: *ICALP 2011, LNCS*, vol. 6756, pp. 222–233. Springer (2011)
19. Zetsche, G.: Silent transitions in automata with storage (2013), to appear in *Proceedings of ICALP 2013*. Available at <http://arxiv.org/abs/1302.3798>.