

# The Algebraic Theory of Parikh Automata

Michaël Cadilhac<sup>1</sup>, Andreas Krebs<sup>2</sup>, and Pierre McKenzie<sup>1,\*</sup>

<sup>1</sup> DIRO at U. de Montréal and Chaire Digiteo ENS Cachan-École Polytechnique  
{cadilhac, mckenzie}@iro.umontreal.ca

<sup>2</sup> WSI at U. Tübingen  
mail@krebs.net

**Abstract.** The Parikh automaton model equips a finite automaton with integer registers and imposes a semilinear constraint on the set of their final settings. Here the theory of typed monoids is used to characterize the language classes that arise algebraically. Complexity bounds are derived, such as containment of the unambiguous Parikh automata languages in  $\text{NC}^1$ . Noting that DetAPA languages are positive supports of rational  $\mathbb{Z}$ -series, DetAPA are further shown stronger than Parikh automata on unary languages. This suggests unary DetAPA languages as candidates for separating the two better known variants of uniform  $\text{NC}^1$ .

## Introduction

The Parikh automaton model was introduced in [19]. It amounts to a nondeterministic finite automaton equipped with registers tallying up the number of occurrences of each transition along an accepting run. Such a run is then deemed successful iff the tuple of final register settings falls within a fixed semilinear set. An *affine* variant of the model in which transitions further induce an affine transformation on the registers was considered in [10]. An *unambiguous* variant of the model was considered in [11]. Tree Parikh automata and other variants were considered in [18].

Recall the tight connection between  $\text{AC}^0$ ,  $\text{ACC}^0$  and  $\text{NC}^1$  and aperiodic monoids, solvable monoids and nonsolvable monoids respectively [2,3]. This connection was refined and studied in depth (see [24] for a lovely account), but the class  $\text{TC}^0 \subseteq \text{NC}^1$  was left out of the picture because the MAJ gate in circuits could not be translated into the operation of a finite algebraic structure. Typed monoids were introduced in [20] as a means of capturing  $\text{TC}^0$  meaningfully in the algebraic framework.

In both the classical and the typed monoid framework, a compelling notion of a natural class of monoids is that of a variety. In both frameworks, different monoid varieties capture different classes of languages as inverse homomorphic images of an accepting subset of the monoid [13,6]. The internal structure of  $\text{NC}^1$  hinges on whether different monoid varieties still capture different classes of languages when the classical notion of a homomorphism is appropriately generalized to capture as above complexity classes such as  $\text{ACC}^0$ ,  $\text{TC}^0$  and  $\text{NC}^1$ .

---

\* Supported by NSERC of Canada and Digiteo.

Our contribution is an algebraic characterization of the language classes defined by the deterministic and unambiguous variants of the Parikh automaton (called CA, for “constrained automaton”) and the affine Parikh automaton. We show:

- the class  $\mathcal{L}_{\text{DetCA}}$  of languages accepted by deterministic CA is the set of languages recognized by typed monoids from  $\mathbf{Z}^+ \wr \mathbf{M}$ , i.e., by wreath products of the monoid of integers with some finite monoid; the least typed monoid variety generated by  $\mathbf{Z}^+ \wr \mathbf{M}$  also captures  $\mathcal{L}_{\text{DetCA}}$
- the class  $\mathcal{L}_{\text{UnCA}}$  of languages accepted by unambiguous CA is the set of languages recognized by typed monoids from  $\mathbf{Z}^+ \square \mathbf{M}$ , i.e., by block products of the monoid of integers with some finite monoid; the least typed monoid variety generated by  $\mathbf{Z}^+ \square \mathbf{M}$  also captures  $\mathcal{L}_{\text{UnCA}}$
- the classes  $\mathcal{L}_{\text{DetAPA}}$  and  $\mathcal{L}_{\text{UnAPA}}$ , of languages accepted by deterministic and by unambiguous affine Parikh automata respectively (where an affine Parikh automaton generalizes the constrained automaton by allowing each transition to perform an affine transformation on the automaton registers), are the Boolean closure of the positive supports of rational series over the integers.

The first two characterizations above add legitimacy to the theory of typed monoids, and they suggest further relevance of that theory to our understanding of  $\text{NC}^1$ . It follows from the characterization of  $\mathcal{L}_{\text{UnCA}}$  that  $\mathcal{L}_{\text{UnCA}} \subseteq \text{NC}^1$ , a fact which is not immediately obvious from the operation of an unambiguous constrained automaton.

The Boolean closure of the class of positive supports of rational series over the integers, hence  $\mathcal{L}_{\text{DetAPA}} = \mathcal{L}_{\text{UnAPA}}$ , can be viewed as a very tightly uniform version of the (DLOGTIME-uniform) class  $\text{PNC}^1$ , introduced in [12] as the log depth analog of the poly time and log space classes PP and PL [16]. Fulfilling  $\text{NC}^1 \subseteq \text{PNC}^1 \subseteq \text{L}$ ,  $\text{PNC}^1$  is robust, pointedly characterized using iterated products of constant dimension integer matrices, but also characterized using paths in bounded width graphs, proof trees in log depth circuits, accepting paths in non-deterministic finite automata or evaluation of a log depth  $\{+, \times\}$ -formula [12]. An elaborate structural complexity evolved around  $\text{PNC}^1$  with the work of [21]. We note that using formal power series as a tool to investigate counting classes below L was already suggested in [1], but with emphasis there on the complexity of performing operations such as inversion and root extraction on such series.

## 1 Preliminaries

*Monoids, integers, vectors.* A monoid is a set  $M$  with an associative operation, usually denoted multiplicatively  $(x, y) \mapsto xy$ , and an identity element denoted 1. For  $S \subseteq M$ , we write  $S^*$  for the monoid generated by  $S$ , i.e., the smallest submonoid of  $M$  containing  $S$ . A (monoid) *morphism* from  $M$  to  $N$  is a map preserving product and identity. Moreover, if  $M = \Sigma^*$  for some alphabet  $\Sigma$  (i.e.,  $\Sigma$  is a finite set of symbols), then  $h$  need only be defined on the elements of  $\Sigma$ .

We write  $\mathbb{N}$ ,  $\mathbb{Z}$ ,  $\mathbb{Z}^+$ ,  $\mathbb{Z}_0^-$  for the sets of nonnegative integers, integers, positive integers, and nonpositive integers respectively. Vectors in  $\mathbb{N}^d$  are noted in bold, e.g.,  $\mathbf{v}$  whose elements are  $v_1, v_2, \dots, v_d$ . We write  $\mathbf{e}_i \in \{0, 1\}^d$  for the vector having a 1 only in position  $i$ , and  $\mathbf{0}$  for the all-zero vector. We view  $\mathbb{N}^d$  as the additive monoid  $(\mathbb{N}^d, +)$ , with  $+$  the component-wise addition and  $\mathbf{0}$  the identity element. We let  $\mathcal{M}_{\mathbb{Z}}(k)$ , for  $k \geq 1$ , be the monoid of square matrices of dimension  $k \times k$  with values in  $\mathbb{Z}$  and with the operation mapping  $(M_1, M_2)$  to  $M_2 M_1$ . In particular, a morphism  $h: \{a, b\}^* \rightarrow \mathcal{M}_{\mathbb{Z}}(k)$  is such that  $h(ab) = h(b) \cdot h(a)$  with  $\cdot$  the usual matrix multiplication. We write  $\Psi_i$  for the projection on the  $i$ -th component,  $\Psi_i(a_1, a_2, \dots, a_i, \dots) = a_i$ .

*Semilinear sets, Parikh image.* A subset  $C$  of  $\mathbb{N}^d$  is *linear* if there exist  $\mathbf{c} \in \mathbb{N}^d$  and a finite  $P \subseteq \mathbb{N}^d$  such that  $C = \mathbf{c} + P^*$ . The subset  $C$  is said to be *semilinear* if it is equal to a finite union of linear sets:  $\{4n + 56 \mid n > 0\}$  is semilinear while  $\{2^n \mid n > 0\}$  is not. We will often use the fact that the semilinear sets are those sets of natural numbers definable in first-order logic with addition [17]. Let  $\Sigma = \{a_1, a_2, \dots, a_n\}$  be an (ordered) alphabet and  $1$  be the empty word. The *Parikh image* is the morphism  $\text{Pkh}: \Sigma^* \rightarrow \mathbb{N}^n$  defined by  $\text{Pkh}(a_i) = \mathbf{e}_i$ , for  $1 \leq i \leq n$  — in particular, we have that  $\text{Pkh}(1) = \mathbf{0}$ . For  $w \in \Sigma^*$ , with  $\text{Pkh}(w) = \mathbf{x}$  and  $a_i \in \Sigma$ , we write  $|w|_{a_i}$  for  $x_i$ . The Parikh image of a language  $L$  is defined as  $\text{Pkh}(L) = \{\text{Pkh}(w) \mid w \in L\}$ . The name of this morphism stems from Parikh's theorem [22], stating that for  $L$  context-free,  $\text{Pkh}(L)$  is semilinear; outside language theory, it is also referred to as the *commutative image*.

*Affine functions.* A function  $f: \mathbb{N}^d \rightarrow \mathbb{N}^d$  is a (total and positive) *affine function* of dimension  $d$  if there exist a matrix  $M \in \mathbb{N}^{d \times d}$  and  $\mathbf{v} \in \mathbb{N}^d$  such that for any  $\mathbf{x} \in \mathbb{N}^d$ ,  $f(\mathbf{x}) = M\mathbf{x} + \mathbf{v}$ . We abusively write  $f = (M, \mathbf{v})$ . We let  $\mathcal{F}_d$  be the monoid of such functions under the operation  $\diamond$  defined by  $(f \diamond g)(\mathbf{x}) = g(f(\mathbf{x}))$ , where the identity element is the identity function, i.e.,  $(Id, \mathbf{0})$  with  $Id$  the identity matrix of dimension  $d$ .

*Automata.* An automaton is a quintuple  $A = (Q, \Sigma, \delta, q_0, F)$  where  $Q$  is a finite set of states,  $\Sigma$  is an alphabet,  $\delta \subseteq Q \times \Sigma \times Q$  is a set of transitions,  $q_0 \in Q$  is the initial state, and  $F \subseteq Q$  is a set of final states. For a transition  $t = (q, a, q') \in \delta$ , define  $\text{From}(t) = q$  and  $\text{To}(t) = q'$ . We define  $\text{Label}_A: \delta^* \rightarrow \Sigma^*$  as the morphism given by  $\text{Label}_A(t) = a$ , with, in particular,  $\text{Label}_A(1) = 1$ , and write  $\text{Label}$  when  $A$  is clear from the context. The set of accepting paths of  $A$ , i.e., the set of words over  $\delta$  describing paths starting from  $q_0$  and ending in  $F$ , is written  $\text{Run}(A)$ . The language of the automaton is  $L(A) = \text{Label}_A(\text{Run}(A))$ . An automaton is *unambiguous* if for all  $w \in L(A)$  there is a unique  $\pi \in \text{Run}(A)$  with  $\text{Label}(\pi) = w$ .

A *constrained automaton (CA)* [10] is a pair  $(A, C)$  where  $A$  is an automaton with  $d$  transitions and  $C \subseteq \mathbb{N}^d$  is semilinear. Its language is  $L(A, C) = \text{Label}_A(\{\pi \in \text{Run}(A) \mid \text{Pkh}(\pi) \in C\})$ . The CA is said to be *deterministic (DetCA)* if  $A$  is deterministic, and *unambiguous (UnCA)* if  $A$  is unambiguous. We write  $\mathcal{L}_{\text{CA}}$ ,  $\mathcal{L}_{\text{DetCA}}$ , and  $\mathcal{L}_{\text{UnCA}}$  for the classes of languages recognized by CA, DetCA, and UnCA, respectively.

An *affine Parikh automaton (APA)* [10] of dimension  $d$  is a triple  $(A, U, C)$  where  $A$  is an automaton with transition set  $\delta$ ,  $U: \delta^* \rightarrow \mathcal{F}_d$  is a morphism, and  $C \subseteq \mathbb{N}^d$  is semilinear. Its language is  $L(A, U, C) = \text{Label}_A(\{\pi \in \text{Run}(A) \mid [U(\pi)](\mathbf{0}) \in C\})$ . The APA is said to be *deterministic (DetAPA)* if  $A$  is deterministic, and *unambiguous (UnAPA)* if  $A$  is unambiguous. We write  $\mathcal{L}_{\text{DetAPA}}$  and  $\mathcal{L}_{\text{UnAPA}}$  for the classes of languages recognized by DetAPA and UnAPA, respectively.

*Transition monoid.* Let  $A = (Q, \Sigma, \delta, q_0, F)$  be a complete deterministic automaton. For  $a \in \Sigma$ , define  $f_a: Q \rightarrow Q$  by  $f_a(q) = q'$  iff  $(q, a, q') \in \delta$ . The *transition monoid*  $M$  of  $A$  is the closure under composition of the set  $\{f_a \mid a \in \Sigma\}$ . The monoid  $M$  acts on  $Q$  naturally by  $q.m = m(q)$ ,  $m \in M$ ,  $q \in Q$ . Write  $\eta: \Sigma^* \rightarrow M$  for the canonical surjective morphism associated, that is, the morphism defined by  $\eta(a) = f_a$ ,  $a \in \Sigma$ . Then  $q.\eta(w)$  is the state reached by reading  $w \in \Sigma^*$  from the state  $q \in Q$ .

## 2 Normal forms of CA and APA

We present several technical lemmata on CA and APA that will help us in devising concise proofs for the algebraic characterizations that follow. Their main purpose is to simplify the constraint set, so that only sign checks on linear combinations of variables are performed.

Recall (e.g., [14]) that for any semilinear set  $C \subseteq \mathbb{Z}^d$ , there is a Boolean combination of expressions of the form:  $\sum_{i=1}^d \alpha_i x_i > c$  and  $\sum_{i=1}^d \alpha_i x_i \equiv_p c$ , with  $\alpha_i, c \in \mathbb{Z}$  and  $p > 1$ , which is true iff  $(x_1, x_2, \dots, x_d) \in C$ . Note that the  $\alpha_i$  may be zero. We define two notions which refine this point of view:

**Definition 1.** *We say that a semilinear set  $C$  is modulo-free if it can be expressed as a Boolean combination of expressions of the form  $\sum_i \alpha_i x_i > c$ , for  $\alpha_i \in \mathbb{Z}$ . We say that  $C$  is basic if it can further be expressed as a positive Boolean combination of expressions of the form  $\sum_i \alpha_i x_i > 0$ .*

The first normal form concerns DetCA and UnCA:

**Lemma 1.** *Every DetCA (resp. UnCA) has the same language  $L \subseteq \Sigma^+$  as another DetCA (resp. UnCA)  $(A, C)$  with  $L(A) = \Sigma^*$  and  $C$  a basic set.*

We also note the following simple fact:

**Lemma 2.** *For  $(A, C_1 \cap C_2)$  a DetCA or an UnCA it holds that:*

$$L(A, C_1 \cap C_2) = L(A, C_1) \cap L(A, C_2) .$$

*The same holds for  $\cup$ .*

We show more in the context of APA to allow the forthcoming proofs of characterization to translate smoothly from CA to APA. In the following, we consider that a matrix  $M \in \mathcal{M}_{\mathbb{Z}}(k)$  is in a set  $C \subseteq \mathbb{Z}^{k^2}$  if the vector consisting of the columns of  $M$  is in  $C$ .

**Lemma 3.** *Let  $L \subseteq \Sigma^+$  be in  $\mathcal{L}_{\text{DetAPA}}$ . There is a morphism  $h: \Sigma^* \rightarrow \mathcal{M}_{\mathbb{Z}}(k)$ , for some  $k$ , and a set  $\mathcal{Z} \subseteq \mathbb{Z}^{k^2}$  expressible as a Boolean combination of expressions  $x_i > 0$ , such that  $L = h^{-1}(\mathcal{Z})$ .*

*Similarly, let  $L \subseteq \Sigma^+$  be in  $\mathcal{L}_{\text{UnAPA}}$ . There is an unambiguous automaton  $A$  with transition set  $\delta$ , a morphism  $h: \delta^* \rightarrow \mathcal{M}_{\mathbb{Z}}(k)$ , for some  $k$ , and a set  $\mathcal{Z} \subseteq \mathbb{Z}^{k^2}$  expressible as a Boolean combination of expressions  $x_i > 0$ , such that  $L = \text{Label}_A(h^{-1}(\mathcal{Z}) \cap \text{Run}(A))$ .*

### 3 Capturing Parikh automata classes algebraically

In this section we characterize DetCA, UnCA, DetAPA, and UnAPA using the theory of (finitely) typed monoids [20].

#### 3.1 Typed monoids

In the following, we will use such notions as language recognition by a finite monoid, varieties of languages, and pseudovarieties of finite monoids (see, e.g., [13]). In algebraic language theory, central tools for finite monoid composition include the *block* and *wreath* products, the definitions of which we recall here, before giving similar definitions in the theory of typed monoids.

Let  $M$  and  $N$  be finite monoids. To distinguish the operation of  $M$  and  $N$ , we denote the operation of  $M$  as  $+$  and its identity element as  $0$  (although this operation is not necessarily commutative) and the operation of  $N$  implicitly and its identity element as  $1$ . A left action of  $N$  on  $M$  is a function mapping pairs  $(n, m) \in N \times M$  to  $nm \in M$  and satisfying  $n(m_1 + m_2) = nm_1 + nm_2$ ,  $n_1(n_2 m) = (n_1 n_2)m$ ,  $n0 = 0$  and  $1m = m$ . Right actions are defined symmetrically. If we have both a right and a left action of  $N$  on  $M$  that further satisfy  $n_1(mn_2) = (n_1 m)n_2$ , we define the bilateral semidirect product  $M ** N$  as the monoid with elements in  $M \times N$  and multiplication defined as  $(m_1, n_1)(m_2, n_2) = (m_1 n_2 + n_1 m_2, n_1 n_2)$ . This operation is associative and  $(0, 1)$  acts as an identity for it. Given only a left action, the unilateral semidirect product  $M * N$  is the bilateral semidirect product  $M ** N$  where the right action on  $M$  is trivial ( $mn = m$ ).

Let  $M, N$  be two monoids. The *wreath product* of  $M$  and  $N$ , written  $M \wr N$ , is defined as the unilateral semidirect product of  $M^N$  and  $N$ , where the left action of  $N$  on  $M^N$  is given by  $(n \cdot f)(n') = f(n'n)$ , for  $f: N \rightarrow M$  and  $n, n' \in N$ . The *block product* of  $M$  and  $N$ , written  $M \square N$ , is defined as the bilateral semidirect product of  $M^{N \times N}$  and  $N$ , where the right (resp. left) action of  $N$  on  $M^{N \times N}$  is given by  $(f \cdot n)(n_1, n_2) = f(n_1, nn_2)$  (resp.  $(n \cdot f)(n_1, n_2) = f(n_1 n, n_2)$ ), for  $f: N \times N \rightarrow M$  and  $n, n_1, n_2 \in N$ .

We now turn to the theory of typed monoids.

**Definition 2 (Typed monoid [20]).** *A typed monoid is a pair  $(S, \mathfrak{S})$  where  $S$  is a finitely generated monoid and  $\mathfrak{S}$  is a finite Boolean algebra of subsets of  $S$  whose elements are called types. We write  $(S, \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_n\})$  for the typed monoid  $(S, \mathfrak{S})$  where  $\mathfrak{S}$  is generated by the  $\mathcal{S}_i$ 's. If  $n = 1$ , we simply write*

$(S, \mathfrak{S}_1)$ . For two typed monoids  $(M, \mathfrak{M})$ ,  $(N, \mathfrak{N})$ , their direct product  $(S, \mathfrak{S}) = (M, \mathfrak{M}) \times (N, \mathfrak{N})$  is defined by  $S = M \times N$ , and  $\mathfrak{S}$  is the Boolean algebra generated by  $\{\mathcal{M} \times \mathcal{N} \mid \mathcal{M} \in \mathfrak{M} \text{ and } \mathcal{N} \in \mathfrak{N}\}$ . A typed monoid  $(S, \mathfrak{S})$  recognizes a language  $L$  if there are a morphism  $h: \Sigma^* \rightarrow S$  and a type  $\mathcal{S} \in \mathfrak{S}$  such that  $L = h^{-1}(\mathcal{S})$ . We write  $\mathcal{L}((S, \mathfrak{S}))$  for the class of languages, over any alphabet, recognized by  $(S, \mathfrak{S})$  and extend this notation naturally to classes of typed monoids.

We view a finite monoid  $M$  as the typed monoid  $(M, 2^M)$ , and write  $\mathbf{M}$  for the class of typed finite monoids; note that the usual notion of language recognition then coincides with the one given here.

The usual wreath product (resp. block product) of  $M$  and  $N$ , i.e., the unilateral (resp. bilateral) semidirect product of  $M^N$  (resp.  $M^{N \times N}$ ) and  $N$ , results, in the infinite monoid case, in monoids with uncountably many elements, failing to fall within the definition of typed monoid. Thus the block product was restricted, in [20], to *type-respecting* functions, that is, functions that only depend on the type of their arguments (multiplied by some constants). Here, we are not concerned with this technicality as all our monoids  $N$  will be finite. Hence we define:

**Definition 3 (Typed block [20] and wreath products).** *Let  $(M, \mathfrak{M})$  and  $(N, \mathfrak{N})$  be typed monoids. The block product (resp. wreath product) of  $(M, \mathfrak{M})$  and  $(N, \mathfrak{N})$ , written  $(M, \mathfrak{M}) \square (N, \mathfrak{N})$  (resp.  $(M, \mathfrak{M}) \wr (N, \mathfrak{N})$ ), is  $(M \square N, \mathfrak{S})$  (resp.  $(M \wr N, \mathfrak{S})$ ) with  $\mathfrak{S} = \{\mathcal{S}_{\mathcal{M}} \mid \mathcal{M} \in \mathfrak{M}\}$  where:*

$$\mathcal{S}_{\mathcal{M}} = \{(f, n) \in S \mid f(1, 1) \in \mathcal{M}\} \text{ (resp. } \mathcal{S}_{\mathcal{M}} = \{(f, n) \in S \mid f(1) \in \mathcal{M}\}) .$$

The appropriateness of typed monoids in the study of the algebraic properties of nonregular languages is witnessed by the following Eilenberg-like theorem of Behle, Krebs, and Reifferscheid:

**Theorem 1 ([6]).** *Varieties of typed monoids and varieties of languages are in a one-to-one correspondence, i.e., (1) Let  $\mathcal{V}$  be a variety of languages and  $\mathbf{V}$  the smallest variety of typed monoids that recognizes all languages in  $\mathcal{V}$ , then  $\mathcal{L}(\mathbf{V}) = \mathcal{V}$ ; (2) Let  $\mathbf{V}$  be a variety of typed monoids and  $\mathbf{W}$  be the smallest variety that recognizes all languages of  $\mathcal{L}(\mathbf{V})$ , then  $\mathbf{V} = \mathbf{W}$ .*

Similar to the untyped algebraic theory of languages, if a typed monoid recognizes a language, it also recognizes its complement. This implies that  $\mathcal{L}_{\text{CA}}$ , which is not closed under complement, does not accept a typed monoid characterization. We will thus focus on characterizing the deterministic and unambiguous classes. Note that we will frequently focus on languages which do not contain the empty word. This is a technical simplification which introduces no loss of generality, as all our typed monoid classes recognize  $\{1\}$  and are closed under union.

### 3.2 Capturing DetCA and UnCA

Let  $\mathbf{Z}^+$  be the set of typed monoids  $\{(\mathbb{Z}, \mathbb{Z}^+)^k \mid k \geq 1\}$ .

**Theorem 2.**  $\mathcal{L}(\mathbf{Z}^+ \wr \mathbf{M}) = \mathcal{L}_{\text{DetCA}}$ .

*Proof.* ( $\mathcal{L}_{\text{DetCA}} \subseteq \mathcal{L}(\mathbf{Z}^+ \wr \mathbf{M})$ ) We first show that  $\mathcal{L}(\mathbf{Z}^+ \wr \mathbf{M})$  is closed under union and intersection. Let  $L_1, L_2 \in \mathcal{L}(\mathbf{Z}^+ \wr \mathbf{M})$  be two languages over  $\Sigma$ , that is, for  $i = 1, 2$ , there exist a finite monoid  $M_i$ , an integer  $k_i$ , a morphism  $h_i: \Sigma^* \rightarrow \mathbb{Z}^{k_i} \wr M_i$ , and a type  $\mathcal{T}_i$  of  $(\mathbb{Z}, \mathbb{Z}^+)^{k_i}$  such that  $L_i = h_i^{-1}(\mathcal{T}_i)$ .

Consider the typed monoid  $(\mathbb{Z}, \mathbb{Z}^+)^{k_1+k_2} \wr (M_1 \times M_2) \in \mathbf{Z}^+ \wr \mathbf{M}$ . This monoid recognizes both the intersection and union of  $L_1$  and  $L_2$  as follows. Define  $h: \Sigma^* \rightarrow \mathbb{Z}^{k_1+k_2} \wr (M_1 \times M_2)$  by  $h(a) = (f_a, (\Psi_2(h_1(a)), \Psi_2(h_2(a))))$  where  $a \in \Sigma$  and  $f_a((m_1, m_2)) = ([\Psi_1(h_1(a))](m_1), [\Psi_1(h_2(a))](m_2)) \in \mathbb{Z}^{k_1+k_2}$ . Now let  $\circ \in \{\cup, \cap\}$ . We define  $\mathcal{T}_\circ = (\mathcal{T}_1 \times \mathbb{Z}^{k_2}) \circ (\mathbb{Z}^{k_1} \times \mathcal{T}_2)$ , and thus  $h^{-1}(\mathcal{T}_\circ) = L_1 \circ L_2$ .

Now let  $(A, C)$  be a DetCA with  $A = (Q, \Sigma, \delta, q_0, F)$ , and suppose (by Lemma 1) that  $F = Q$  and that the constraint set is expressed by a positive Boolean combination of clauses of the form  $\sum_{t \in \delta} \alpha_t x_t > 0$ . Closure of  $\mathcal{L}(\mathbf{Z}^+ \wr \mathbf{M})$  under  $\cup$  and  $\cap$  together with Lemma 2 imply that it is enough to argue the case in which  $C$  is defined by a single such clause.

Let  $M$  be the transition monoid of  $A$ ,  $\eta: \Sigma^* \rightarrow M$  the canonical morphism associated. We now define  $h: \Sigma^* \rightarrow \mathbb{Z} \wr M$  as follows. Let  $\tau: M \times \Sigma \rightarrow \delta$  be defined by  $\tau(m, a) = (q_0 \cdot m, a, q_0 \cdot m \eta(a))$ . Then:

$$h(a) = (f_a, \eta(a)), \text{ where } f_a(m) = \alpha_{\tau(m,a)} .$$

Now let  $w = w_1 w_2 \cdots w_n \in \Sigma^*$  and  $\pi = \pi_1 \pi_2 \cdots \pi_n$  where  $w_i \in \Sigma$  and  $\pi_i \in \delta$  for every  $1 \leq i \leq n$ , such that  $\pi$  is the unique accepting path in  $A$  from  $q_0$  labeled  $w$ . We have:

$$\begin{aligned} h(w) &= (f_{w_1} + \eta(w_1) \cdot f_{w_2} + \cdots + \eta(w_1 w_2 \cdots w_{n-1}) \cdot f_{w_n}, \eta(w)) \\ [\Psi_1(h(w))](\eta(1)) &= \alpha_{\tau(\eta(1), w_1)} + \sum_{i=2}^n \alpha_{\tau(\eta(w_1 \cdots w_{i-1}), w_i)} , \end{aligned}$$

note that  $q_0 \cdot \eta(w_1 \cdots w_{i-1})$  is  $\text{From}(\pi_i)$  and thus  $\tau(\eta(w_1 \cdots w_{i-1}), w_i) = \pi_i$ , hence:

$$[\Psi_1(h(w))](\eta(1)) = \sum_{i=1}^n \alpha_{\pi_i} = \sum_{t \in \delta} |\pi|_t \times \alpha_t .$$

Thus,  $\text{Pkh}(\pi) \in C$  iff  $[\Psi_1(h(w))](\eta(1)) > 0$ . Hence with the type  $\mathcal{T} = \{(f, m) \in (\mathbb{Z}, \mathbb{Z}^+) \wr M \mid f(\eta(1)) > 0\}$ , which is indeed a type of  $(\mathbb{Z}, \mathbb{Z}^+) \wr M$ , we have that  $h^{-1}(\mathcal{T}) = L(A, C)$ .

$(\mathcal{L}(\mathbf{Z}^+ \wr \mathbf{M}) \subseteq \mathcal{L}_{\text{DetCA}})$  Let  $L \subseteq \Sigma^*$  be recognized by  $(\mathbb{Z}, \mathbb{Z}^+)^k \wr M$  using a type  $\mathcal{T}$  and a morphism  $h: \Sigma^* \rightarrow (\mathbb{Z}^k)^M \times M$ , and write for convenience  $h_i(w) = \Psi_i(h(w))$ ,  $i = 1, 2$ . Let  $A$  be the automaton  $(M, \Sigma, \delta, 1, M)$ , where:

$$\delta = \{(m, a, m') \mid m \in M, a \in \Sigma \text{ and } m' = m \cdot h_2(a)\} .$$

Now as  $\mathcal{L}_{\text{DetCA}}$  is closed under union and intersection, we may suppose that the type  $\mathcal{T}$  is of the following form:

$$\mathcal{T} = \prod_{i=1}^k \{(f, m) \mid f(1) \in \mathcal{T}_i\} ,$$

where each  $\mathcal{T}_i \in \{\emptyset, \mathbb{Z}_0^-, \mathbb{Z}^+, \mathbb{Z}\}$ . Define  $T = \mathcal{T}_1 \times \mathcal{T}_2 \times \cdots \times \mathcal{T}_k$ , and the semilinear set  $C$  consisting of elements:

$$(x_{t_1}, x_{t_2}, \dots, x_{t_{|\delta|}}) \text{ s.t. } \sum_{t \in \delta} x_t \times [h_1(\text{Label}(t))](\text{From}(t)) \in T .$$

We claim that the language of the DetCA  $(A, C)$  is  $L$ . Let  $w = w_1 w_2 \cdots w_n \in \Sigma^*$ . There is an (accepting) path in  $A$  labeled  $w$  going through the states  $1 = h_2(1), h_2(w_1), h_2(w_1 w_2), \dots, h_2(w_1 w_2 \cdots w_n)$ . Thus the sum computed by the semilinear set is  $h_1(w_1) + h_2(w_1) \cdot h_1(w_2) + \cdots + h_2(w_1 w_2 \cdots w_n) \cdot h_1(w_n)$ , taken at the point 1. This is precisely  $[h_1(w)](1)$ , and thus checking whether it belongs to  $T$  is equivalent to checking whether  $h(w) \in \mathcal{T}$ . Hence  $L = L(A, C)$ .  $\square$

Now  $\mathcal{L}_{\text{DetCA}}$  is a variety of languages and we may naturally ask whether the smallest variety containing  $\mathbf{Z}^+ \wr \mathbf{M}$ , which recognizes only the languages of  $\mathcal{L}_{\text{DetCA}}$  by Theorem 1, is closed under iterated wreath product. We note this is not the case. Let  $U_1 = (\{0, 1\}, \times)$ , then:

**Theorem 3.** *There is a language  $L \notin \mathcal{L}_{\text{CA}}$  recognized by  $U_1 \wr (\mathbb{Z}, \mathbb{Z}^+)$  and by  $(\mathbb{Z}, \mathbb{Z}^+) \wr (\mathbb{Z}, \mathbb{Z}^+)$ .*

We now turn to unambiguous CA:

**Theorem 4.**  $\mathcal{L}(\mathbf{Z}^+ \square \mathbf{M}) = \mathcal{L}_{\text{UnCA}}$ .

*Proof.* ( $\mathcal{L}_{\text{UnCA}} \subseteq \mathcal{L}(\mathbf{Z}^+ \square \mathbf{M})$ ) We first note that  $\mathcal{L}(\mathbf{Z}^+ \square \mathbf{M})$  is closed under union and intersection; this is the same proof as in Theorem 2 except that  $f_a$  is now defined as:

$$f_a((m_1, m_2), (m'_1, m'_2)) = ([\Psi_1(h_1(a))](m_1, m'_1), [\Psi_1(h_2(a))](m_2, m'_2)) .$$

Next consider an UnCA  $(A, C)$  with  $A = (Q, \Sigma, \delta, q_0, F)$ , and suppose (using Lemma 1) that  $L(A) = \Sigma^*$  and that the constraint set is expressed by a positive Boolean combination of clauses of the form  $\sum_{t \in \delta} \alpha_t x_t > 0$ . Closure of  $\mathcal{L}(\mathbf{Z}^+ \square \mathbf{M})$  under  $\cup$  and  $\cap$  together with Lemma 2 imply that it is enough to argue the case in which  $C$  is defined by a single such clause.

Let  $M$  be the transition monoid of the deterministic version of  $A$ , obtained using the powerset construction. Let  $A'$  be defined as  $A$  with all transitions inverted (i.e.,  $(p, a, q)$  is in  $A$  iff  $(q, a, p)$  is in  $A'$ ). Let  $M'$  be the transition monoid of the deterministic version of  $A'$ , using again the powerset construction, and let  $M^c$  be the monoid defined on the same elements as  $M'$  but with the operation reversed (i.e.,  $m_1 \circ_{M'} m_2$  in  $M'$  is  $m_2 \circ_{M^c} m_1$  in  $M^c$ ; this is still a



monoid as  $\circ_{M^c}$  is still associative). We will show that  $L(A, C)$  is recognized by  $(S, \mathfrak{S}) = (\mathbb{Z}, \mathbb{Z}^+) \square (M \times M^c)$ .

Write  $\eta$  and  $\eta^c$  for the canonical morphisms associated with  $M$  and  $M^c$ ; for  $m \in M$  and  $R \subseteq Q$ , write  $R.m$  for the action of  $m$  on  $R$ , and likewise for  $M^c$ . We first note that for  $w \in \Sigma^*$ ,  $\{q_0\}.\eta(w)$  is the set of states of  $A$  that can be reached in  $A$  reading  $w$  from  $q_0$ , and, likewise, that  $F.\eta^c(w)$  is the set of states in  $A$  from which reading  $w$  leads to a final state.

Now for  $m_1 \in M$ ,  $a \in \Sigma$ , and  $m_2 \in M^c$ , let  $\tau(m_1, a, m_2)$  be the unique transition in  $A$  from a state in  $\{q_0\}.m_1$  to a state in  $F.m_2$  labeled  $a$ . We show that  $\tau$  is well-defined. Let  $w_1, w_2$  such that  $\eta(w_1) = m_1$  and  $\eta^c(w_2) = m_2$ ; this means that there are  $w_1$ -labeled paths in  $A$  from  $q_0$  to *any* state in  $\{q_0\}.m_1$ , and, likewise,  $w_2$ -labeled paths in  $A$  from *any* state in  $F.m_2$  to a final state. (*Existence*): as  $w_1aw_2$  is in  $\Sigma^* = L(A)$ , there is a transition in  $A$  from a state in  $\{q_0\}.m_1$  to a state in  $F.m_2$  labeled  $a$ . (*Uniqueness*): if two transitions  $(p, a, p')$  and  $(q, a, q')$  are such that  $p, q \in \{q_0\}.m_1$  and  $p', q' \in F.m_2$ , this means that there are multiple accepting paths in  $A$  labeled  $w_1aw_2$ , contradicting the unambiguity of  $A$ . We now define the morphism  $h: \Sigma^* \rightarrow S$  by:

$$h(a) = (f_a, (\eta(a), \eta^c(a))), \text{ where}$$

$$f_a((m_1, m_2), (m'_1, m'_2)) = \alpha_{\tau(m_1, a, m'_2)} .$$

Now let  $w = w_1w_2 \cdots w_n \in \Sigma^*$ ,  $w_i \in \Sigma$  for every  $1 \leq i \leq n$ , and  $\pi$  be the unique path in  $A$  from  $q_0$  to a final state labeled  $w$ . Then:

$$\pi = \pi_1\pi_2 \cdots \pi_n \quad \text{where}$$

$$\pi_i = \tau(\eta(w_1w_2 \cdots w_{i-1}), w_i, \eta^c(w_{i+1}w_{i+2} \cdots w_n)) ,$$

and thus:

$$[\Psi_1(h(w))](\eta(1), \eta^c(1)) = \sum_{t \in \delta} |\pi|_t \times \alpha_t .$$

Thus  $\text{Pkh}(\pi) \in C$  iff  $[\Psi_1(h(w))](\eta(1), \eta^c(1)) > 0$ . Hence with the type  $S = \{(f, m) \in S \mid f((\eta(1), \eta^c(1))) \in \mathbb{Z}^+\}$ , which is indeed a type in  $\mathfrak{S}$  as  $\mathbb{Z}^+$  is a type of  $(\mathbb{Z}, \mathbb{Z}^+)$ , we have that  $h^{-1}(S) = L(A, C)$ .

$(\mathcal{L}(\mathbb{Z}^+ \square \mathbf{M}) \subseteq \mathcal{L}_{\text{UNCA}})$  Let  $L \subseteq \Sigma^*$  be recognized by  $(\mathbb{Z}, \mathbb{Z}^+)^k \square M$  using a type  $\mathcal{T}$  and a morphism  $h: \Sigma^* \rightarrow \mathbb{Z}^{M \times M} \times M$ , and write for convenience  $h_i(w) = \Psi_i(h(w))$ ,  $i = 1, 2$ . For any  $(s_1, s_2) \in M \times M$ , Let  $A(s_1, s_2)$  be the automaton  $(M \times M, \Sigma, \delta, (s_1, s_2), M \times \{1\})$  where:

$$\delta = \{((m_1, m_2), a, (m'_1, m'_2)) \mid$$

$$m'_1 = m_1h_2(a) \text{ and } h_2(a)m'_2 = m_2 \in M \text{ and } a \in \Sigma\} .$$

Note that  $w \in L(A(s_1, s_2))$  implies  $h_2(w) = s_2$ . We argue that  $A(s_1, s_2)$  is unambiguous for any  $(s_1, s_2) \in M \times M$ . We show that for any  $w \in \Sigma^*$  and any  $(s_1, s_2) \in M \times M$ ,  $w$  is the label of at most one accepting path in  $A(s_1, s_2)$ , by induction on  $|w|$ . If  $w = 1$ , then every  $A(s_1, s_2)$  has at most one accepting path labeled  $w$ . Now let  $w = a \cdot v$  for  $v \in \Sigma^*$ . Suppose  $w \in L(A(s_1, s_2))$ . This implies

that  $h_2(w) = s_2$ . The states that can be reached from  $(s_1, h_2(w))$  reading  $a$  are all of the form  $(s_1 h_2(a), m)$ ,  $m \in M$ . Now  $v$  should be accepted by the automaton  $A$  where the initial state is set to one of these states; thus there is only one state fitting,  $(s_1 h_2(a), h_2(v))$ . By induction hypothesis, there is only one path in  $A(s_1 h_2(a), h_2(v))$  recognizing  $v$ , thus there is only one path in  $A(s_1, h_2(w))$  recognizing  $w$ . This shows that for any  $s_1, s_2$ ,  $A(s_1, s_2)$  is unambiguous.

Now, with  $e = (1, 1)$ , and as  $\mathcal{L}_{\text{UnCA}}$  is closed under union and intersection, we may suppose that the type  $\mathcal{T}$  is of the following form:

$$\mathcal{T} = \prod_{i=1}^k \{(f, m) \mid f(e, e) \in \mathcal{T}_i\} ,$$

where each  $\mathcal{T}_i \in \{\emptyset, \mathbb{Z}_0^-, \mathbb{Z}^+, \mathbb{Z}\}$ . Define  $T = \mathcal{T}_1 \times \mathcal{T}_2 \times \cdots \times \mathcal{T}_k$ , and the semilinear set  $C$  consisting of elements:

$$(x_{t_1}, x_{t_2}, \dots, x_{t_{|\delta|}}) \text{ s.t. } \sum_{t \in \delta} x_t \times [h_1(\text{Label}(t))] (\Psi_1(\text{From}(t)), \Psi_2(\text{To}(t))) \in T .$$

We show that  $\bigcup_{m \in M} L(A(1, m), C)$  is  $L$ . Let  $w = w_1 w_2 \cdots w_n \in \Sigma^*$ . There is a unique accepting path in  $A(1, h_2(w))$  (and in no other  $A(1, m)$ ) labeled  $w$ , and it is going successively through the states  $(h_2(1), h_2(w)) = (1, h_2(w))$ ,  $(h_2(w_1), h_2(w_2 \cdots w_n))$ ,  $\dots$ ,  $(h_2(w), 1) = (h_2(w), h_2(1))$ . For this path, the sum computed by the semilinear set is:

$$\sum_{i=1}^n h_2(w_1 \cdots w_{i-1}) \cdot h_1(w_i) \cdot h_2(w_{i+1} \cdots w_n) ,$$

at the point  $(1, 1)$ . This is precisely  $[h_1(w)](1, 1)$ , and checking whether it is in  $T$  amounts to checking whether  $h(w) \in \mathcal{T}$ , thus  $L = \bigcup_{m \in M} L(A(1, m), C)$ .  $\square$

We derive an interesting property of the logical characterization and circuit complexity of UnCA. Let  $\text{MSO}[\langle] be the monadic second-order logic with  $\langle$  as the unique numerical predicate, and  $\text{FO}+\text{G}[\langle$  be the first-order logic with group quantifiers and  $\langle$  as the unique numerical predicate. Both logics express exactly the regular languages (these are respectively the classical results of Büchi [8] and Barrington, Immerman, Straubing [4]). Now define the *extended majority* quantifier  $\widehat{\text{Maj}}$ , introduced in [5], as:  $w \models \widehat{\text{Maj}} x \langle \varphi_i \rangle_{i=1, \dots, m}$  iff  $\sum_{j=1}^{|w|} |\{i \mid w_{x=j} \models \varphi_i\}| - |\{i \mid w_{x=j} \not\models \varphi_i\}| > 0$ . Then:$

**Corollary 1.** *A language is in  $\mathcal{L}_{\text{UnCA}}$  iff it can be expressed as a Boolean combination of formulas of the form:*

$$\widehat{\text{Maj}} x \langle \varphi_i \rangle_{i=1, \dots, m}$$

where each  $\varphi_i$  is an  $\text{MSO}[\langle$  formula or an  $\text{FO}+\text{G}[\langle$  formula. Hence,  $\mathcal{L}_{\text{UnCA}} \subsetneq \text{NC}^1$ .

### 3.3 Capturing DetAPA and UnAPA

Write  $\mathfrak{Z}^+(k)$  for the type set of  $(\mathbb{Z}, \mathbb{Z}^+)^k$ , that is, the sets expressible as a Boolean combination of expressions of the form  $x_i > 0$ . Let  $\mathbf{ZMat}^+$  be the set of typed monoids  $\{(\mathcal{M}_{\mathbb{Z}}(k), \mathfrak{Z}^+(k \times k)) \mid k \geq 1\}$ , then:

**Theorem 5.**  $\mathcal{L}(\mathbf{ZMat}^+) = \mathcal{L}_{\text{DetAPA}}$ .

*Proof.* ( $\mathcal{L}_{\text{DetAPA}} \subseteq \mathcal{L}(\mathbf{ZMat}^+)$ ) This is a direct consequence of Lemma 3.

( $\mathcal{L}(\mathbf{ZMat}^+) \subseteq \mathcal{L}_{\text{DetAPA}}$ ) Given  $k \geq 1$ , a type  $\mathcal{Z}$  of  $(\mathbb{Z}, \mathbb{Z}^+)^{k \times k}$ , and a morphism  $h: \Sigma^* \rightarrow \mathcal{M}_{\mathbb{Z}}(k)$ , we build a two-state DetAPA of dimension  $k^2$  for  $h^{-1}(\mathcal{Z})$ . First, let  $h': \Sigma^* \rightarrow \mathcal{M}_{\mathbb{Z}}(k^2)$  be such that  $h'(a)$  is the Kronecker product of the identity matrix of dimension  $k$  and  $h(a)$ . Define  $\mathbf{e} = (\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_k)$  where each  $\mathbf{e}_i$  is of dimension  $k$ . Then for any word  $w$ ,  $h(w) \in \mathcal{Z}$  iff  $h'(w)\mathbf{e} \in \mathcal{Z}$ . Now let  $A = (\{r, s\}, \Sigma, \delta, r, \{s\})$ , with  $\delta = \{r, s\} \times \Sigma \times \{s\}$ . Then let  $U: \delta^* \rightarrow \mathcal{F}_{k^2}$  for  $q \in \{r, s\}$ ,  $a \in \Sigma$ , and  $\mathbf{x} \in \mathbb{Z}^{k^2}$  be defined by:

$$[U((q, a, s))](\mathbf{x}) = \begin{cases} h'(a)\mathbf{e} & \text{if } q = r, \\ h'(a)\mathbf{x} & \text{otherwise.} \end{cases}$$

This implies that for  $w \in \Sigma^+$  and  $\pi$  its unique accepting path in  $A$ , it holds that  $[U(\pi)](\mathbf{0}) = h'(w)\mathbf{e}$ . Thus  $L(A, U, \mathcal{Z}) = h^{-1}(\mathcal{Z})$ .  $\square$

**Theorem 6.**  $\mathcal{L}(\mathbf{ZMat}^+ \square \mathbf{M}) = \mathcal{L}_{\text{UnAPA}}$ .

*Proof.*  $\mathcal{L}_{\text{UnAPA}} \subseteq \mathcal{L}(\mathbf{ZMat}^+ \square \mathbf{M})$  is the same as  $\mathcal{L}_{\text{UnCA}} \subseteq \mathcal{L}(\mathbf{Z}^+ \square \mathbf{M})$  in Theorem 4, thanks to Lemma 3.

$\mathcal{L}(\mathbf{ZMat}^+ \square \mathbf{M}) \subseteq \mathcal{L}_{\text{UnAPA}}$  is the same as  $\mathcal{L}(\mathbf{Z}^+ \square \mathbf{M}) \subseteq \mathcal{L}_{\text{UnCA}}$  in Theorem 4 for the automaton part, and the same as Theorem 5 for the constraint set and affine function parts.  $\square$

Now, applying the same arguments as in [9, Lemma 5], we have that DetAPA can simulate unambiguity, and thus  $\mathcal{L}_{\text{UnAPA}} = \mathcal{L}_{\text{DetAPA}}$ . This translates nicely in the algebraic framework thanks to Theorem 1:

**Theorem 7.** *The smallest variety containing  $\mathbf{ZMat}^+ \square \mathbf{M}$  is equal to that containing  $\mathbf{ZMat}^+$ .*

## 4 Formal power series

In this section, we show that the languages of DetAPA are those expressible as a Boolean combination of positive supports of  $\mathbb{Z}$ -valued rational series. This helps us derive a separation over the unary languages between  $\mathcal{L}_{\text{CA}}$  and  $\mathcal{L}_{\text{DetAPA}}$  — the separation was known ([10, Proposition 28]), but not over unary languages.

**Definition 4 (e.g., [7]).** *Functions from  $\Sigma^*$  into  $\mathbb{Z}$  are called ( $\mathbb{Z}$ -)series. For such a series  $r$ , it is customary to write  $(r, w)$  for  $r(w)$ . We write  $\text{supp}_+(r)$  for the positive support of  $r$ , i.e.,  $\{w \mid (r, w) > 0\}$ .*

A linear representation of dimension  $k \geq 1$  is a triple  $(\mathbf{s}, h, \mathbf{g})$  such that  $\mathbf{s} \in \mathbb{Z}^k$  is a row vector,  $\mathbf{g} \in \mathbb{Z}^k$  is a column vector, and  $h: \Sigma^* \rightarrow \mathbb{Z}^{k \times k}$  is a monoid morphism, where the operation of the matrix monoid is the usual matrix multiplication. It defines the series  $r = \|\langle \mathbf{s}, h, \mathbf{g} \rangle\|$  with  $(r, w) = \mathbf{s}h(w)\mathbf{g}$ .

A series is said to be rational if it is defined by a linear representation. We write  $\mathbb{Z}^{\text{rat}}\langle\langle \Sigma^* \rangle\rangle$  for the set of rational series.

For a class  $\mathcal{C}$  of languages, write  $\text{BC}(\mathcal{C})$  for the Boolean closure of  $\mathcal{C}$ . Arguments similar to those used in proving Theorem 5 allow us to show:

**Theorem 8.** *Over any alphabet  $\Sigma$ ,  $\mathcal{L}_{\text{DetAPA}} = \text{BC}(\text{supp}_+(\mathbb{Z}^{\text{rat}}\langle\langle \Sigma^* \rangle\rangle))$ .*

*Proof.* ( $\mathcal{L}_{\text{DetAPA}} \subseteq \text{BC}(\text{supp}_+(\mathbb{Z}^{\text{rat}}\langle\langle \Sigma^* \rangle\rangle))$ ) First note that there is a rational series  $r$  such that  $\text{supp}_+(r) = \{1\}$ . Let  $L$  be in  $\mathcal{L}_{\text{DetAPA}}$ ; we may thus suppose that  $1 \notin L$ . By the same token as in the proof of Theorem 5, there is a morphism  $h: \Sigma^* \rightarrow \mathcal{M}_{\mathbb{Z}}(k)$ , for some  $k$ , a vector  $\mathbf{v} \in \{0, 1\}^k$ , and a type  $\mathcal{Z}$  of  $(\mathbb{Z}, \mathbb{Z}^+)^k$  such that:

$$L = \{w \mid h(w)\mathbf{v} \in \mathcal{Z}\} .$$

Further, similar to Lemma 2,  $L(A, U, C_1 \circ C_2) = L(A, U, C_1) \circ L(A, U, C_2)$ , for  $\circ \in \{\cup, \cap\}$  and any DetAPA  $(A, U, C_1 \circ C_2)$ . Moreover,  $L(A, U, \bar{C}) = \overline{L(A, U, C)} \cap L(A)$ . We may thus suppose that  $\mathcal{Z}$  is reduced to  $\mathbb{Z}^{i-1} \times \mathbb{Z}^+ \times \mathbb{Z}^{k-i}$  for some  $i$ .

Now let  $h'$  be the morphism from  $\Sigma^*$  to  $\mathbb{Z}^{k \times k}$  (with the *usual* matrix multiplication as operation), where  $h'(a) = (h(a))^\text{T}$ , with  $a \in \Sigma$  and  $M^\text{T}$  the transpose of  $M$ . Note that  $h(a_1 a_2) = h(a_2)h(a_1) = ((h(a_1))^\text{T}(h(a_2))^\text{T})^\text{T}$ , which is  $(h'(a_1 a_2))^\text{T}$ ; more generally,  $h(w) = (h'(w))^\text{T}$ . Thus we have that  $\mathbf{v}^\text{T}h'(w) = (h(w)\mathbf{v})^\text{T}$ . Hence with  $\mathbf{s} = \mathbf{v}^\text{T}$  and  $\mathbf{g}$  the column vector  $\mathbf{e}_i$ ,  $\mathbf{s}h'(w)\mathbf{g} > 0$  iff  $h(w)\mathbf{v} \in \mathcal{Z}$ .

Now the triple  $(\mathbf{s}, h', \mathbf{g})$  is a linear representation of a rational series which associates  $w$  to  $\mathbf{s}h'(w)\mathbf{g}$ , and this concludes the proof.

$(\text{BC}(\text{supp}_+(\mathbb{Z}^{\text{rat}}\langle\langle \Sigma^* \rangle\rangle)) \subseteq \mathcal{L}_{\text{DetAPA}})$  As  $\mathcal{L}_{\text{DetAPA}}$  is closed under union, complement, and intersection, we need only show that  $\text{supp}_+(\mathbb{Z}^{\text{rat}}\langle\langle \Sigma^* \rangle\rangle) \subseteq \mathcal{L}_{\text{DetAPA}}$ .

Let  $(\mathbf{s}, h, \mathbf{g})$  be a linear representation of dimension  $k$  of a rational series  $r$  over the alphabet  $\Sigma$ . Define  $h': \Sigma^* \rightarrow \mathcal{M}_{\mathbb{Z}}(k)$  by letting  $h'(a) = (h(a))^\text{T}$ , for  $a \in \Sigma$ . Then for  $w \in \Sigma^*$ ,  $h(w) = (h'(w))^\text{T}$ . Now the rest of the proof is similar to that of Theorem 5: define  $A = (\{r, t\}, \Sigma, \delta, r, \{r, t\})$ , with  $\delta = \{r, t\} \times \Sigma \times \{t\}$ . Then let  $U: \delta^* \rightarrow \mathcal{F}_k$  for  $q \in \{r, t\}$ ,  $a \in \Sigma$ , and  $\mathbf{x} \in \mathbb{Z}^k$ , be defined by:

$$[U((q, a, t))](\mathbf{x}) = \begin{cases} h'(a)\mathbf{s} & \text{if } q = r, \\ h'(a)\mathbf{x} & \text{otherwise.} \end{cases}$$

This implies that for  $w \in \Sigma^*$  and  $\pi$  its unique accepting path in  $A$ , it holds that  $[U(\pi)](\mathbf{0}) = \mathbf{s}h(w)$ . Thus letting  $C = \{\mathbf{x} \mid \mathbf{x}\mathbf{g} > 0\}$ , with  $\mathbf{x}$  a row vector and  $\mathbf{g}$  a column vector, we have that  $L(A, U, C) = \text{supp}_+(r)$ .  $\square$

*Remark 1.* The class of positive supports of  $\mathbb{Z}$ -rational series is the class of  $\mathbb{Q}$ -stochastic languages (see, e.g., [23]). As we are interested in showing that  $\mathcal{L}_{\text{DetAPA}}$  is not closed under concatenation, it is worth noting that  $\mathbb{Q}$ -stochastic languages are not closed under concatenation. We mention three proofs of this fact. Two proofs [15,23] show that  $\mathbb{Q}$ -stochastic languages are not closed under concatenation with a *finite* language; such a concatenation is expressible as a finite union of  $\mathbb{Q}$ -stochastic languages, and is thus not directly applicable to our case. A third proof [25] shows that the  $\mathbb{Q}$ -stochastic language  $L = \{a^i \# (a + \#)^* \# a^i \mid i \in \mathbb{N}\}$  is such that  $L \cdot \{a, \#\}^*$  is not  $\mathbb{Q}$ -stochastic. We conjecture that  $L \cdot \{a, \#\}^*$  is neither in  $\mathcal{L}_{\text{DetAPA}}$ , but the proof given in [25] does not apply directly to our case. Finally, we note that the fact that unary  $\mathbb{Q}$ -stochastic languages are not closed under union [23] implies, as any regular language is  $\mathbb{Q}$ -stochastic, that there are nonregular unary languages in  $\mathcal{L}_{\text{DetAPA}}$ .

Let  $\#\text{NC}^1$  be the class of functions computed by DLOGTIME-uniform arithmetic circuits of polynomial size and logarithmic depth and  $\text{PNC}^1$  be the class of languages expressible as  $\{w \mid f(w) > 0\}$  for  $f \in \#\text{NC}^1$  (see [12]). Note that this class is included in L. As iterated matrix multiplication can be done in  $\#\text{NC}^1$  and  $\text{PNC}^1$  is closed under the Boolean operations, it is readily seen from Theorem 8 that:

**Corollary 2.**  $\mathcal{L}_{\text{DetAPA}} \subseteq \text{PNC}^1$ .

## Conclusion

Connections between variants of the Parikh automaton and complexity classes were investigated. In particular, natural characterizations of the language classes defined by deterministic and unambiguous constrained automata, in the theory of typed monoids, were obtained. We hope that these characterizations will suggest refinements that may help to better understand classes such as  $\text{PNC}^1$  and  $\text{NC}^1$ .

We note in conclusion that the unary languages in  $\mathcal{L}_{\text{DetAPA}}$ , and indeed the bounded languages in  $\mathcal{L}_{\text{DetAPA}}$ , can be shown to belong to the DLOGTIME-DCL-uniform variant of  $\text{NC}^1$ . Recall that the latter is not known to equal what is commonly referred to as DLOGTIME-uniform  $\text{NC}^1$  (see [26, p. 162]), or ALOGTIME. Yet we were unable to show that the unary languages in  $\mathcal{L}_{\text{DetAPA}}$  belong to the latter. Do they?

## References

1. Allender, E., Arvind, V., Mahajan, M.: Arithmetic complexity, Kleene closure, and formal power series. *Theory Comput. Syst.* 36(4), 303–328 (2003)
2. Barrington, D.A.M.: Bounded-width polynomial size branching programs recognize exactly those languages in  $\text{NC}^1$ . *J. Computer and System Sciences* 38, 150–164 (1989)

3. Barrington, D.A.M., Thérien, D.: Finite monoids and the fine structure of  $NC^1$ . *J. Association of Computing Machinery* 35, 941–952 (1988)
4. Barrington, D.A.M., Immerman, N., Straubing, H.: On uniformity within  $NC^1$ . *J. Comput. Syst. Sci* 41(3), 274–306 (1990)
5. Behle, C., Krebs, A., Mercer, M.: Linear circuits, two-variable logic and weakly blocked monoids. In: *Mathematical Foundations of Computer Science. LNCS*, vol. 4708, pp. 147–158. Springer-Verlag (2007)
6. Behle, C., Krebs, A., Reifferscheid, S.: Typed monoids - an Eilenberg-like theorem for non regular languages. In: *Algebraic Informatics. LNCS*, vol. 6742, pp. 97–114. Springer (2011)
7. Berstel, J., Reutenauer, C.: *Noncommutative Rational Series with Applications. Encyclopedia of Mathematics and its Applications*, Cambridge University Press (2010)
8. Büchi, J.R.: Weak second order arithmetic and finite automata. *Z. Math. Logik Grundlagen Math.* 6, 66–92 (1960)
9. Cadilhac, M., Finkel, A., McKenzie, P.: Bounded Parikh automata. In: *Proc. 8th Internat. Conference on Words. EPTCS*, vol. 63, pp. 93–102 (2011)
10. Cadilhac, M., Finkel, A., McKenzie, P.: Affine Parikh automata. *RAIRO - Theoretical Informatics and Applications* 46(04), 511–545 (2012)
11. Cadilhac, M., Finkel, A., McKenzie, P.: Unambiguous constrained automata. In: *Developments in Language Theory. LNCS*, vol. 7410, pp. 239–250. Springer Berlin / Heidelberg (2012)
12. Caussinus, H., McKenzie, P., Thérien, D., Vollmer, H.: Nondeterministic  $NC^1$  computation. *J. Computer and System Sciences* 57, 200–212 (1998)
13. Eilenberg, S.: *Automata, Languages, and Machines, Volume B. Pure and Applied Mathematics*, Academic Press (1976)
14. Enderton, H.B.: *A Mathematical Introduction to Logic*. Academic Press (1972)
15. Fliess, M.: Propriétés booléennes des langages stochastiques. *Mathematical Systems Theory* 7(4), 353–359 (1974)
16. Gill, J.: Computational complexity of probabilistic turing machines. *SIAM J. Computing* 6, 675–695 (1977)
17. Ginsburg, S., Spanier, E.H.: Semigroups, Presburger formulas and languages. *Pacific J. Mathematics* 16(2), 285–296 (1966)
18. Karianto, W.: Parikh automata with pushdown stack. Diploma thesis, RWTH Aachen (2004)
19. Klaedtke, F., Rueß, H.: Monadic second-order logics with cardinalities. In: *30th Int. Coll. Automata, Languages and Programming*. pp. 681–696 (2003)
20. Krebs, A., Lange, K.J., Reifferscheid, S.: Characterizing  $TC^0$  in terms of infinite groups. *Theory of Computing Systems* 40(4), 303–325 (2007)
21. Limaye, N., Mahajan, M., Rao, B.V.R.: Arithmetizing classes around  $NC^1$  and  $L$ . *Theory Comput. Syst.* 46(3), 499–522 (2010)
22. Parikh, R.J.: On context-free languages. *J. ACM* 13(4), 570–581 (1966)
23. Salomaa, A., Soittola, M.: *Automata-Theoretic Aspects of Formal Power Series*. Springer, New York (1978)
24. Straubing, H.: *Finite Automata, Formal Logic, and Circuit Complexity*. Birkhäuser, Boston (1994)
25. Turakainen, P.: Some closure properties of the family of stochastic languages. *Information and Control* 18(3), 253–256 (1971)
26. Vollmer, H.: *Introduction to Circuit Complexity – A Uniform Approach. Texts in Theoretical Computer Science*, Springer Verlag (1999)