class of *bounded semilinear languages*, defined as the class of bounded languages L for which there exists a socle $w_1, \ldots, w_n$ such that $\mathsf{Iter}_{(w_1,\ldots,w_n)}(L)$ is semilinear; in particular, the Parikh image of a language in BSL is semilinear. Note that the converse does not hold, as $\{a^{2^n} b a^m \mid m, n \in \mathbb{N}\}$ is bounded and has a semilinear Parikh image, but is not in BSL.

Regular bounded languages can be characterized by a subclass of regular expressions. Let Σ be an alphabet. A *semilinear*[c] *regular expression* (SLRE) [9] is a finite set of *branches* on alphabet Σ, defined as expressions of the form $y_1 x_1^* y_2 x_2^* \cdots y_n x_n^* y_{n+1}$, where $x_i \in \Sigma^+$ and $y_i \in \Sigma^*$. The language of an SLRE is the union of the languages of each of its branches. A regular language is bounded iff it is expressible as a SLRE [13].

Automata. We then fix our notation about automata. An automaton is a quintuple $A = (Q, \Sigma, \delta, q_0, F)$ where Q is the finite set of states, Σ is an alphabet, $\delta \subseteq Q \times \Sigma \times Q$ is the set of transitions, $q_0 \in Q$ is the initial state and $F \subseteq Q$ are the final states. As usual, an automaton $A = (Q, \Sigma, \delta, q_0, F)$ induces a finite directed labeled graph $G_A = (Q, \delta)$ where Q is the set of nodes and δ is the set of (labeled) arcs. For a transition (or a labeled arc) $t = (q, a, q') \in \delta$, we define $\mathsf{From}(t) = q$ and $\mathsf{To}(t) = q'$. Moreover, we define $\mu_A \colon \delta^* \to \Sigma^*$ to be the morphism given by $\mu_A(t) = a$, with, in particular, $\mu_A(\varepsilon) = \varepsilon$, and we write μ when A is clear from the context. A *path* π on A is a word $\pi = t_1 \cdots t_n \in \delta^*$ such that $\mathsf{To}(t_i) = \mathsf{From}(t_{i+1})$ for $1 \le i < n$; we extend From and To to paths, letting $\mathsf{From}(\pi) = \mathsf{From}(t_1)$ and $\mathsf{To}(\pi) = \mathsf{To}(t_n)$. We say that $\mu(\pi)$ is the *label* of π. A path π is said to be *accepting* if $\mathsf{From}(\pi) = q_0$ and $\mathsf{To}(\pi) \in F$; we write $\mathsf{Run}(A)$ for the language over δ of accepting paths on A. We write $L(A)$ for the language of A, i.e., the labels of the accepting paths. The automaton A is said to be *deterministic* if $((p, a, q) \in \delta \wedge (p, a, q') \in \delta)$ implies $q = q'$. An ε-automaton is an automaton $A = (Q, \Sigma, \delta, q_0, F)$ as above, except with $\delta \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times Q$ so that in particular μ_A becomes an erasing morphism.

Flat and restricted flat automata. For an ε-automaton $A = (Q, \Sigma, \delta, q_0, F)$, several notions of *flatness* have been defined in the literature and we wish to specify our usage of the word here. A *cycle* in A is a path $\pi \in \delta^+$ such that $\mathsf{From}(\pi) = \mathsf{To}(\pi)$. An *elementary* cycle in A is a cycle π in which the only repeated state is the initial (and final) state $\mathsf{From}(\pi)$. Our notion of flatness is the following: the automaton A is *flat* if no two elementary cycles in A share a state. This definition is equivalent to those in [2, 8].

Note that if A is flat, then A induces a natural directed acyclic graph D_A on the vertex set $\{[q] : q \in Q\}$, where $[q]$ is the set $\{q\}$ together with all the states reachable from q along an elementary cycle: there is an arc between $[q]$ and $[q']$, with $[q] \neq [q']$, iff there are $p \in [q]$ and $p' \in [q']$ such that there is a transition between p and p' in A. We introduce a (proper) subclass of flat automata: an ε-automaton

[c]The usage of *semilinear* here is not directly related to its usage in *semilinear set*.

$A = (Q, \Sigma, \delta, q_0, \{q_f\})$ is *rflat* (for *restricted flat*) if A is flat and D_A is a straight line from $[q_0]$ to $[q_f]$.

The following properties are easy to see:

(i) No nested cycles occur in a flat automaton;
(ii) If A is flat then $\mathsf{Run}(A)$ is a regular bounded language on δ (hence SLRE);
(iii) Any regular bounded language on δ (hence SLRE) is $\mathsf{Run}(A)$ for some flat A;
(iv) If A is rflat then $\mathsf{Run}(A)$ is expressible as a branch on δ;
(v) Any language of a branch on δ is $\mathsf{Run}(A)$ for some rflat A;
(vi) A is rflat iff it is *restricted simple* in the sense of [4].

Rational transduction. Let Σ and T be two alphabets. Let A be an automaton over the alphabet $(\Sigma \cup \{\varepsilon\}) \times (\mathrm{T} \cup \{\varepsilon\})$, where the concatenation is defined by $(u_1, v_1).(u_2, v_2) = (u_1u_2, v_1v_2)$. Then A defines the *rational transduction* τ_A from languages L on Σ to languages on T given by $\tau_A(L) = \{v \in \mathrm{T}^* \mid (\exists u \in L)[(u, v) \in L(A)]\}$. Closure under rational transduction for a class $\mathcal{C}$ is the property that for any language $L \in \mathcal{C}$ and any automaton A, $\tau_A(L) \in \mathcal{C}$. We say that τ_A is a *deterministic* rational transduction if A is deterministic with respect to the first component of its labels, i.e., if $(p, (a, b), q)$ and $(p, (a, b'), q')$ are transitions of A, then $b = b'$ and $q = q'$.

Affine functions. We consider the vectors in $\mathbb{N}^d$ to be *column* vectors. A function $f \colon \mathbb{N}^d \to \mathbb{N}^d$ is a (total and positive) *affine function* of dimension d if there exist a matrix $M \in \mathbb{N}^{d \times d}$ and $\overline{v} \in \mathbb{N}^d$ such that for any $\overline{x} \in \mathbb{N}^d$, $f(\overline{x}) = M.\overline{x} + \overline{v}$. We note $f = (M, \overline{v})$ and write $\mathcal{F}_d$ for the set of such functions; we view $\mathcal{F}_d$ as the monoid $(\mathcal{F}_d, \diamond)$ with $(f \diamond g)(\overline{x}) = g(f(\overline{x}))$, where the identity element of the monoid is the identity function. Let U be a monoid morphism from Σ^* to $\mathcal{F}_d$. For $w \in \Sigma^*$, we write U_w for $U(w)$, so that the application of $U(w)$ to a vector $\overline{v}$ is written $U_w(\overline{v})$, and U_ε is the identity function. We define $\mathcal{M}(U)$ as the multiplicative matrix monoid generated by the matrices used to define U, i.e., $\mathcal{M}(U) = \{M \mid (\exists a \in \Sigma)(\exists \overline{v})[U_a = (M, \overline{v})]\}^*$.

3. Parikh Automata and Constrained Automata

The following notations will be used in defining Parikh finite word automata (PA) formally. Let Σ be an alphabet, $d \in \mathbb{N}^+$, and D a finite subset of $\mathbb{N}^d$. Following [17], let $\Psi \colon (\Sigma \times D)^* \to \Sigma^*$ and $\tilde{\Phi} \colon (\Sigma \times D)^* \to \mathbb{N}^d$ be two morphisms defined, for $\ell = (a, \overline{v}) \in \Sigma \times D$, by $\Psi(\ell) = a$ and $\tilde{\Phi}(\ell) = \overline{v}$. The function Ψ is called the *projection on* Σ and the function $\tilde{\Phi}$ is called the *extended Parikh image*. As an example, for a word $\omega \in \{(a_i, \overline{e_i}) \mid 1 \le i \le n\}^*$, the value of $\tilde{\Phi}(\omega)$ is the Parikh image of $\Psi(\omega)$.

Definition 1 (Parikh automaton [17]) *Let Σ be an alphabet, $d \in \mathbb{N}^+$, and D a finite subset of $\mathbb{N}^d$. A* Parikh automaton (PA) *of dimension d over $\Sigma \times D$ is a pair*

(A, C) where A is a finite automaton over $\Sigma \times D$, and $C \subseteq \mathbb{N}^d$ is a semilinear set. The PA language is $L(A, C) = \{\Psi(\omega) \mid \omega \in L(A) \wedge \tilde{\Phi}(\omega) \in C\}$.

The PA is said to be deterministic (DetPA) *if for every state q of A and every $a \in \Sigma$ there exists at most one pair $(q', \overline{v})$ with q' a state and $\overline{v} \in D$ such that $(q, (a, \overline{v}), q')$ is a transition of A. The PA is said to be* rflat *if A is rflat. We write $\mathcal{L}_{PA}$ (resp. $\mathcal{L}_{DetPA}$) for the class of languages recognized by PA (resp. DetPA).*

In [5], PA are characterized by the following simpler model:

Definition 2 (Constrained automaton [5]) *A* constrained automaton (CA) *over an alphabet Σ is a pair (A, C) where A is a finite automaton over Σ with d transitions, and $C \subseteq \mathbb{N}^d$ is a semilinear set. Its language is $L(A, C) = \{\mu(\pi) \mid \pi \in \mathsf{Run}(A) \wedge \Phi(\pi) \in C\}$.*

The CA is said to be deterministic (DetCA) *if A is deterministic. An ε-CA is defined as a CA except that A is an ε-automaton. Finally, the CA is said to be* rflat *if A is rflat.*

Theorem 3. (i) *PA and CA define the same class of languages* [5]*;*
(ii) *DetPA and DetCA define the same class of languages* [5]*;*
(iii) *ε-CA and CA (and thus PA) define the same class of languages;*
(iv) *Rflat DetPA and rflat DetCA define the same class of languages.*

Proof. Parts (i) and (ii) are proved in [5]. Part (iii) follows from (i) and the closure of the class of PA languages under erasing morphisms [17]. To prove part (iv), we argue that the proof of (i) and (ii) appearing in [5] applies verbatim.

In one direction, let (A, C) be a DetPA where A is rflat. Define B as the automaton A in which the vector-part of the labels is dropped: a transition $(p, (a, \overline{v}), q)$ in A appears as (p, a, q) in B and write $h \colon \delta_A \to \delta_B$ this correspondence. Note that h is a 1-1 correspondence between the transitions of A and B thanks to the rflatness of A: indeed, for two transitions t_1, t_2 in A to get merged by h, they should be of the form $t_1 = (p, (a, \overline{v_1}), q)$ and $t_2 = (p, (a, \overline{v_2}), q)$. Suppose that $t_1 \neq t_2$, hence $\overline{v_1} \neq \overline{v_2}$, and this would render A non rflat; we deduce that h is injective and it is surjective by construction, hence h is a bijection. This 1-1 correspondence shows in particular that B is rflat, as G_A and G_B are the same. Moreover (A, C) is a DetPA, thus it describes a deterministic automaton with respect to the first component of the labels of A, hence B is deterministic. With $\{t_1, \ldots, t_n\}$ the set of transitions of B, define D as the following semilinear set:

$$(x_1, \ldots, x_n) \in D \quad \Leftrightarrow \quad \sum_{i=1}^{n} x_i \times \tilde{\Phi}(\mu_A(h^{-1}(t_i))) \in C.$$

Then (B, D) is a rflat DetCA with language $L(A, C)$.

In the other direction, let (A, C) be a DetCA where $A = (Q, \Sigma, \delta, q_0, F)$ is rflat. Write $\delta = \{t_1, \ldots, t_n\}$. Define B as the automaton with state set Q and with a transition $(q, (a, \Phi(t_i)), q')$ for each transition $t_i = (q, a, q')$ of A. Then B is rflat, as it has the same graph as A, and (B, C) is a DetPA, as B is deterministic with respect to the first component of its labels. Finally, the language of B is:

$$L(B) = \{\omega \mid (\exists \pi \in \mathsf{Run}(A))[\Psi(\omega) = \mu_A(\pi) \wedge \tilde{\Phi}(\omega) = \Phi(\pi)]\},$$

hence $L(B, C) = L(A, C)$. $\square$

Finally, we note that a related model has been defined and used in the context of model checking:

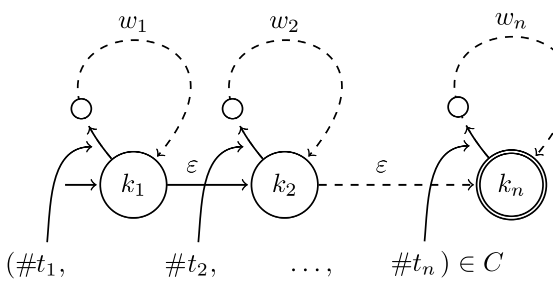
Definition 4 ([4]) *A* 1-*CQDD is a finite set of rflat DetCA. Its language is the union of the languages of each DetCA. We write* $\mathcal{L}_{1\text{-}CQDD}$ *for the class of languages recognized by* 1-*CQDD.*

4. Bounded Parikh Automata

Let $\mathcal{L}_{\text{BoundedPA}}$ be the set $\mathcal{L}_{\text{PA}} \cap$ BOUNDED of bounded PA languages, and similarly let $\mathcal{L}_{\text{BoundedDetPA}}$ be $\mathcal{L}_{\text{DetPA}} \cap$ BOUNDED.

Theorem 6 below characterizes $\mathcal{L}_{\text{BoundedPA}}$ as the class BSL of bounded semilinear languages.[d] In one direction of the proof, given $L \in$ BSL, an ε-CA for L is constructed. We describe this simple construction here:

Construction 5 (Canonical ε-CA for $w_1, \ldots, w_n$ subject to $C \subseteq \mathbb{N}^n$) Let $w_1, \ldots, w_n \in \Sigma^+$ be given words and $C \subseteq \mathbb{N}^n$ be a semilinear set. We describe a PA for the language $\{w_1^{i_1} w_2^{i_2} \cdots w_n^{i_n} \mid (i_1, \ldots, i_n) \in C\}$. Informally, the automaton A will consist of n elementary cycles labeled $w_1, \ldots, w_n$ which do not share any state, and traversed at their origins by a single ε-labeled path from k_1 leading to a unique final state k_n. Then the semilinear constraint E will be defined to monitor $(\#t_1, \ldots, \#t_n)$ in accordance with C, where t_i is the first transition of the cycle for w_i and $\#t_i$ is the number of occurrences of t_i in a run of A. Graphically:



Formally, let $k_j = \sum_{1 \leq i < j} |w_i|$, $1 \leq j \leq n+1$, with, in particular, $k_1 = 0$, and set $Q = \{0, 1, \ldots, k_{n+1} - 1\}$. Then A is the ε-automaton $(Q, \Sigma, \delta, q_0, F)$ where $q_0 = k_1$,

[d] This result can be deduced from the recent result of Ibarra and Seki [15]. However, we need its forthcoming proof to provide some corollaries.

$F = \{k_n\}$ and for any $1 \leq i < n$, there is a transition $(k_i, \varepsilon, k_{i+1})$ and for any $1 \leq i \leq n$ an elementary cycle $t_i\rho_i$ labeled w_i through the states $k_i, k_i+1, \ldots, k_{i+1}-1, k_i$, where t_i is a transition and ρ_i a path. Then $E \subseteq \mathbb{N}^{|\delta|}$ is the semilinear set defined by $(\#t_1, \ldots, \#t_2, \ldots, \ldots, \#t_n, \ldots) \in E$ iff $(\#t_1, \#t_2, \ldots, \#t_n) \in C$. $\square$

Theorem 6. $\mathcal{L}_{BoundedPA} = \mathrm{BSL}$.

Proof. ($\mathcal{L}_{\mathrm{BoundedPA}} \subseteq \mathrm{BSL}$.) Let $L \subseteq \Sigma^*$ be a bounded language of $\mathcal{L}_{\mathrm{PA}}$, and $w_1, \ldots, w_n$ be a socle of L. Define $E = \mathsf{Iter}_{(w_1,\ldots,w_n)}(L)$. Let $\mathrm{T} = \{a_1, \ldots, a_n\}$ be a fresh alphabet ($\mathrm{T} \cap \Sigma = \emptyset$), and let $h\colon \mathrm{T}^* \to \Sigma^*$ be the morphism defined by $h(a_i) = w_i$. Then the language $L' = h^{-1}(L) \cap (a_1^* \cdots a_n^*)$ is in $\mathcal{L}_{\mathrm{PA}}$ by closure of $\mathcal{L}_{\mathrm{PA}}$ under inverse morphism and intersection [17]. But $\Phi(L') = E$, and as any language of $\mathcal{L}_{\mathrm{PA}}$ has a semilinear Parikh image [17], E is semilinear. Thus the iteration set E of the bounded language L with respect to its socle $w_1, \ldots, w_n$ is semilinear, and this is the meaning of L belonging to BSL.

($\mathrm{BSL} \subseteq \mathcal{L}_{\mathrm{BoundedPA}}$.) Let $L \in \mathrm{BSL}$. Of course $L \in \mathrm{BOUNDED}$. Let $w_1, \ldots, w_n$ be a socle of L such that $C = \mathsf{Iter}_{(w_1,\ldots,w_n)}(L)$ is semilinear. We leave out the simple proof that L equals the language of the "canonical ε-CA for $w_1, \ldots, w_n$ subject to C" of Construction 5. Since ε-CA and PA capture the same languages by Theorem 3, $L \in \mathcal{L}_{\mathrm{PA}}$. $\square$

Theorem 6 and the known closure properties of BOUNDED and $\mathcal{L}_{\mathrm{PA}}$ imply:

Corollary 7. BSL *is closed under union, intersection, concatenation, and morphism.*

Proof. Let $L_1, L_2 \in \mathrm{BSL}$. By Theorem 6, both languages are in $\mathcal{L}_{\mathrm{PA}}$. Moreover, $\mathcal{L}_{\mathrm{PA}}$ is closed under union, intersection, concatenation, and morphism [17], BOUNDED is closed under union, intersection, concatenation, and morphism [11]. This implies that $L_1 \cup L_2$, $L_1 \cap L_2$, L_1L_2, and $h(L_1)$ are all bounded languages in $\mathcal{L}_{\mathrm{PA}}$, and by Theorem 6, are all in BSL. $\square$

We note, in the same vein, that although BSL is not closed under inverse morphism (e.g., with h the all-erasing morphism on $\{a, b\}^*$, we have $h^{-1}(\{\varepsilon\}) = \{a, b\}^*$, which is not bounded) we have:

Corollary 8. BSL *is closed under inverse morphism followed by the intersection with a language in* BSL.

Proof. Let $L_1, L_2 \in \mathrm{BSL}$. By Theorem 6, both languages are in $\mathcal{L}_{\mathrm{PA}}$. Moreover, $\mathcal{L}_{\mathrm{PA}}$ is closed under intersection and inverse morphism [17], and the intersection of any language with a bounded language is a bounded language [11]. This implies that $h^{-1}(L_1) \cap L_2$ is a bounded language in $\mathcal{L}_{\mathrm{PA}}$, and by Theorem 6, in BSL. $\square$

Moreover, Theorem 6 helps in showing that if the iteration set of a bounded language w.r.t. *one* of its socles is semilinear, then for *every* socle of the language, the iteration set of the language w.r.t. that socle is semilinear:

Corollary 9. BSL *is the set of bounded languages which have* all *of their iteration sets semilinear.*

Proof. If a bounded language has *all* of its iteration sets semilinear, then it is in BSL. For the reverse inclusion, note that the first half of the proof of Theorem 6 shows that for any socle of a language in $\mathcal{L}_{\mathrm{BoundedPA}}$, the iteration set of the language w.r.t. this socle is semilinear. As $\mathcal{L}_{\mathrm{BoundedPA}} = \mathrm{BSL}$, all the iteration sets of a language in BSL are semilinear. □

Finally, note that the iteration sets of a BSL language are semilinear sets with a special form, which depends on a socle: they contain *all* the possible ways to iterate the words of the socle to obtain a word in the language. Thus defining a bounded language "using" a semilinear set does not directly show that it is in BSL; e.g., $C = \{(x, y) \mid x \text{ is even} \wedge y \in \mathbb{N}\}$ is a semilinear set defining the language a^* using the socle a, a, and yet $\mathsf{Iter}_{(a,a)}(a^*) \neq C$, so C is not a semilinear iteration set of a^*, and we may not directly conclude that $a^* \in \mathrm{BSL}$. However, Theorem 6 provides an easy proof that if a bounded language is defined "using" a semilinear set, then its iteration sets w.r.t. any other prescribed socle are computable semilinear sets:

Corollary 10. *Let $w_1, \ldots, w_n \in \Sigma^*$ and $C \subseteq \mathbb{N}^n$ be a semilinear set. Then $L = \{w_1^{i_1} \cdots w_n^{i_n} \mid (i_1, \ldots, i_n) \in C\}$ is in BSL. Also, for any given socle $w'_1, \ldots, w'_m$ of L, the iteration set of L w.r.t. $w'_1, \ldots, w'_m$ is a semilinear set that we can compute.*

Proof. First, Construction 5 for $w_1, \ldots, w_n$ subject to C provides an ε-CA for L. This does not directly show that $L \in \mathrm{BSL}$, since C is not, in the general case, an iteration set (i.e., $\mathsf{Iter}(L)$ for some socle). However, this shows that L is a bounded language of $\mathcal{L}_{\mathrm{PA}}$, and thus, by Theorem 6, it is a language of BSL.

For the second part, we follow the first half of the proof of Theorem 6 and check that all the operations are computable. So we construct the morphism h which maps a_i to w'_i for $1 \leq i \leq m$. The closure properties of PA being effective [17], we can construct a PA for $L' = h^{-1}(L) \cap (a_1^* \cdots a_m^*)$. Finally, the Parikh image of L' is a semilinear set that we can compute [17], concluding the proof, as this set is the iteration set of L w.r.t. $w'_1, \ldots, w'_m$. □

To the best of our knowledge, Corollary 10 provides the first effective method to obtain the iteration set, w.r.t. a prescribed socle, of a bounded semilinear language described using a semilinear set.

5. Bounded Parikh Automata are Determinizable

Parikh automata cannot be made deterministic in general. Indeed, Klaedtke and Rueß [17] have shown that $\mathcal{L}_{\mathrm{DetPA}}$ is closed under complement while $\mathcal{L}_{\mathrm{PA}}$ is not, so that $\mathcal{L}_{\mathrm{DetPA}} \subsetneq \mathcal{L}_{\mathrm{PA}}$, and [5] further exhibits languages witnessing the separation. In this section, we show that PA can be determinized when their language is bounded. The purpose of this section is to show:

Theorem 11. *Every $L \in \mathcal{L}_{BoundedPA}$ is the union of the languages of rflat DetCA.*

This implies:

Corollary 12. BSL $= \mathcal{L}_{BoundedPA} = \mathcal{L}_{BoundedDetPA} = \mathcal{L}_{1\text{-}CQDD}$.

Proof. The first equality is Theorem 6. For the second, we have that $\mathcal{L}_{\mathrm{BoundedDetPA}} \subseteq \mathcal{L}_{\mathrm{BoundedPA}}$; for the converse, if $L \in \mathcal{L}_{\mathrm{BoundedPA}}$ then it is the union of the languages of rflat DetCA by Theorem 11, and as $\mathcal{L}_{\mathrm{DetPA}}$ is closed under union, $L \in \mathcal{L}_{\mathrm{BoundedDetPA}}$. For the third, we note that rflat DetCA, and thus 1-CQDD, recognize only bounded languages, thus $\mathcal{L}_{\text{1-CQDD}} \subseteq \mathcal{L}_{\mathrm{BoundedPA}}$; for the converse, Theorem 11 states that $\mathcal{L}_{\mathrm{BoundedPA}} \subseteq \mathcal{L}_{\text{1-CQDD}}$. □

We show Theorem 11 in two steps. First, in Section 5.1, we note that the canonical ε-CA of Construction 5 has a crucial property which we call "constraint-determinism," i.e., the fact that the nondeterminism of the automaton is not used in the constraint set (formal definitions follow). We show that CA with this property are naturally expressed with a model of one-way deterministic automaton which allows for some counter manipulation: a restricted version of the *deterministic affine PA* introduced in [5]. Second, in Section 5.2, we show that any bounded language accepted by such a device is a finite union of languages of rflat DetCA. We then conclude the proof of Theorem 11 in Section 5.3.

5.1. *From constraint-deterministic CA to deterministic affine PA*

We first formally define the property of constraint-determinism:

Definition 13 (Constraint-determinism) *A CA (A, C) is said to be* constraint-deterministic *if no two paths π_1 and π_2 in $\mathsf{Run}(A)$ for which $\mu_A(\pi_1) = \mu_A(\pi_2)$ can be distinguished by C. Formally:*

$$(\forall \pi_1, \pi_2 \in \mathsf{Run}(A)) \quad \mu_A(\pi_1) = \mu_A(\pi_2) \Rightarrow (\Phi(\pi_1) \in C \leftrightarrow \Phi(\pi_2) \in C)\,.$$

Given a constraint-deterministic CA (A, C), we will consider the deterministic version of A and follow, within it, the paths traced in A. To this purpose, we will need a model which allows for some simple counter manipulations; we propose the *affine Parikh automaton*, that we introduced and studied in [5], as the right model for this task, as it allows for the needed expressiveness while providing a nice mathematical framework for the proofs.

Intuitively, an affine Parikh automaton will be defined as a finite automaton that also operates on a tuple of counters. Each transition of the automaton will blindly apply an affine function to the tuple of counters. A word w will be deemed accepted by the affine Parikh automaton iff some accepting run of the finite automaton on w also has the cumulative effect of transforming the tuple of counters, initially $\overline{0}$, to a tuple belonging to a prescribed semilinear set.

Definition 14 (Affine Parikh automaton [5]) *An* affine Parikh automaton (APA) *of dimension d is a triple (A, U, C) where A is an automaton with transition set δ, U is a morphism from δ^* to $\mathcal{F}_d$, and $C \subseteq \mathbb{N}^d$ is a semilinear set. The language of the APA is $L(A, U, C) = \{\mu(\pi) \mid \pi \in \mathsf{Run}(A) \wedge U_\pi(\overline{0}) \in C\}$.*

The APA is said to be deterministic (DetAPA) *if A is. We write $\mathcal{L}_{\mathrm{APA}}$ (resp. $\mathcal{L}_{DetAPA}$) for the class of languages recognized by APA (resp. DetAPA).*

The APA is said to be finite-monoid *if $\mathcal{M}(U)$ is finite; this is not the general case.*

In other terms, APA are automata equipped with d counters $c_1, \ldots, c_d$, and each transition computes some action $c_i \leftarrow k_i + \sum_j a_{i,j} \times c_j$ on the d counters. One interesting class of finite-monoid APA, and the one we will focus on, is when *no sum of counter is allowed*, i.e., when all $a_{i,j}$ are either 0 or 1, and if $a_{i,j}$ is 1, then for all $j' \neq j$, $a_{i,j'}$ is 0.

Lemma 15. *Any ε-CA (A, C) having the constraint-determinism property has the same language as a finite-monoid DetAPA (A', U, E) such that $L(A) = L(A')$.*

Proof. We outline the idea before giving the details. Let (A, C) be the ε-CA. We first apply the standard subset construction and obtain a deterministic automaton $\underline{A}$ equivalent to A. Consider a state $\underline{q}$ of $\underline{A}$. Suppose that after reading some word w leading $\underline{A}$ into state $\underline{q}$ we had, for each $q \in \underline{q}$, the Parikh image $\overline{c_{w,q}}$ (counting transitions in A, i.e., recording the occurrences of each transition in A) of *some* initial w-labeled path leading A into state q. Suppose that $(\underline{q}, a, \underline{r})$ is a transition in $\underline{A}$. How can we compute, for each $r \in \underline{r}$, the Parikh image $\overline{c_{wa,r}}$ of *some* initial wa-labeled path leading A into r? It suffices to pick any $q \in \underline{q}$ for which some a-labeled path leads A from q to r (possibly using the ε-transitions in A) and to add to $\overline{c_{w,q}}$ the contribution of this a-labeled path. A DetAPA transition on a is well-suited to mimic this computation, since an affine transformation can first "flip" the current Parikh q-count tuple "over" to the Parikh r-count tuple and then add to it the q-to-r contribution. Hence a DetAPA $(\underline{A}, \cdot, \cdot)$ upon reading a word w leading to its state $\underline{q}$ is able to keep track, for each $q \in \underline{q}$, of the Parikh image of *some* initial w-labeled path leading A into q. We need constraint-determinism only to reach the final conclusion: if a word w leads $\underline{A}$ into a final state $\underline{q}$, then some $q \in \underline{q}$ is final in A, and because of constraint-determinism, imposing membership in C for the Parikh image of the particular initial w-labeled path leading A to q kept track of by the DetAPA is as good as imposing membership in C for the Parikh image of any other initial w-labeled path leading A to q.

We now give the details. Say $A = (Q, \Sigma, \delta, q_0, F)$, and identify Q with $\{1, \ldots, |Q|\}$. For $p, q \in Q$ and $a \in \Sigma$ define $p \overset{a}{\rightsquigarrow} q$ to be a shortest path from p to q labeled by a — lexicographically smallest among shortest paths, for definiteness, as its length can be greater than one because of ε-transitions, — or $\perp$ if none exists. Let $\underline{A} = (2^Q, \Sigma, \underline{\delta}, \underline{q_0}, \underline{F})$ be the deterministic version of A defined by $\underline{q_0} = \{q_0\}$, $\underline{F} = \{\underline{q} \mid \underline{q} \cap F \neq \emptyset\}$, and:

$$\underline{\delta} = \{(\underline{p}, a, \underline{q}) \mid q \in \underline{q} \leftrightarrow (\exists p \in \underline{p})[p \overset{a}{\rightsquigarrow} q \neq \perp]\}.$$

We have that $L(\underline{A}) = L(A)$. Note that, by construction, for any path π in A from q_0 to q, there exists a path $\underline{\pi}$ in $\underline{A}$ from $\underline{q_0}$ to a state $\underline{q}$ such that $q \in \underline{q}$ and $\mu_A(\pi) = \mu_{\underline{A}}(\underline{\pi})$.

We now attach an affine function to each transition of $\underline{A}$, where the functions are of dimension $(|Q|.|\delta| + 1)$. We first define $V : \underline{\delta}^* \to \mathcal{F}_{|Q|.|\delta|}$, and will later add the extra component. We write V_π for $V(\pi)$. The intuition is as follows. Consider a path $\underline{\pi}$ on $\underline{A}$ from the initial state to a state $\underline{q}$ — the empty path is considered to be from $\underline{q_0}$ to $\underline{q_0}$. We view $V_{\underline{\pi}}(\overline{0})$ as a list of counters $(\overline{c_1}, \ldots, \overline{c_{|Q|}})$ where $\overline{c_q} \in \mathbb{N}^{|\delta|}$. We will ensure that for any $q \in \underline{q}$, $\overline{c_q}$ is the Parikh image of a path π in A from q_0 to q such that $\mu_A(\pi) = \mu_{\underline{A}}(\underline{\pi})$. If two such paths π_1 and π_2 exist, we may choose one arbitrarily, as they are equivalent in the following sense: if ρ is such that $\pi_1\rho \in \mathsf{Run}(A)$ and $\Phi(\pi_1\rho) \in C$, then the same holds for π_2.

For $\underline{p} \subseteq Q$, $q \in Q$, and $a \in \Sigma$, let $P(\underline{p}, q, a)$ be the smallest $p \in \underline{p}$ such that $p \overset{a}{\rightsquigarrow} q \neq \perp$ (we will consider only cases where at least one such p exists). Let $\underline{t} = (\underline{p}, a, \underline{q})$ be a transition of $\underline{A}$. We define $V_{\underline{t}}$ such that for $q \in \overline{q}$ and $p = P(\overline{p}, q, a)$, the application of $V_{\underline{t}}$ sets $\overline{c_q}$ to $\overline{c_p} + \Phi(p \overset{a}{\rightsquigarrow} q)$. Formally:

$$V_{\underline{t}} = \left(\sum_{q \in \underline{q}} M\big(P(\underline{p}, q, a),\ q\big), \qquad \sum_{q \in \underline{q}} N\big(q,\ \Phi(P(\underline{p}, q, a) \overset{a}{\rightsquigarrow} q)\big) \right)$$

where $M(p, q)$ is the matrix which transfers the p-th counter to the q-th, and zeroes the others, and $N(q, \overline{d})$ is the shift of $\overline{d} \in \mathbb{N}^{|\delta|}$ to the q-th counter. More precisely, $M(p, q)_{i,j} = 1$ iff there exists $1 \leq e \leq |\delta|$ such that $i = (q-1).|\delta| + e$ and $j = (p-1).|\delta| + e$; likewise, $N(q, \overline{d}) = (0^{(q-1)\cdot|\delta|}) \cdot \overline{d} \cdot (0^{(|Q|-q)\cdot|\delta|})$. The matrices appearing in V are 0-1 matrices with at most one nonzero entry per row; composing such matrices preserves this property, thus $\mathcal{M}(V)$ is finite.

Assertion 16. *Let $\underline{\pi}$ be a path on $\underline{A}$ from $\underline{q_0}$ to some state $\underline{q}$. Let $(\overline{c_1}, \ldots, \overline{c_{|Q|}}) = V_{\underline{\pi}}(\overline{0})$, where $\overline{c_q} \in \mathbb{N}^{|\delta|}$. Then for all $q \in \underline{q}$, $\overline{c_q}$ is the Parikh image of a path in A from q_0 to q labeled by $\mu(\underline{\pi})$.*

We show Assertion 16 by induction. If $|\underline{\pi}| = 0$, then $\mathsf{To}(\pi) = \{q_0\}$ and $\overline{c_{q_0}}$ is by definition all-zero. Thus $\overline{c_{q_0}}$ is the Parikh image of the empty path from and to q_0 in A. Let $\underline{\pi}$ be such that $|\underline{\pi}| > 0$, and consider a state $q \in \mathsf{To}(\underline{\pi})$. Write $\underline{\pi} = \underline{\rho} \cdot \underline{t}$, with $\underline{t} \in \underline{\delta}$, and let $p = P(\mathsf{To}(\underline{\rho}), q, \mu(t))$ and $\zeta = p \overset{\mu(t)}{\rightsquigarrow} q$. The induction hypothesis asserts that the p-th counter of $V_{\underline{\rho}}(\overline{0})$ is the Parikh image of a path ρ on A from q_0

to p labeled by $\mu(\underline{\rho})$. Thus, the q-th counter of $V_{\underline{\pi}}(\overline{0})$ is $\Phi(\rho) + \Phi(\zeta)$, which is the Parikh image of $\rho\zeta$, a path from q_0 to q labeled by $\mu(\underline{\pi})$. This concludes the proof of Assertion 16.

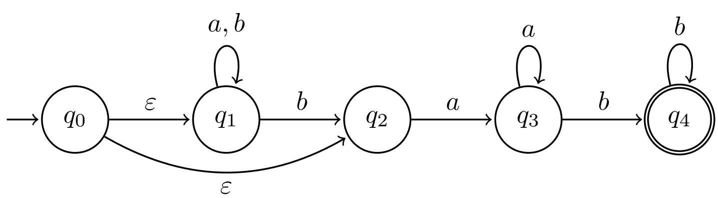
We now define $U \colon \underline{\delta}^* \to \mathcal{F}_{|Q|.|\delta|+1}$. We add a component to the functions of V, such that for $\underline{\pi} \in \mathsf{Run}(\underline{A})$, the last component of $U_{\underline{\pi}}(\overline{0})$ is 0 if $\mathsf{To}(\underline{\pi}) \cap F = \emptyset$ and $\min(\mathsf{To}(\underline{\pi}) \cap F)$ otherwise. For $\underline{t} = (\underline{p}, a, \underline{q}) \in \underline{\delta}$, let:

$$U_{\underline{t}} : (\overline{x}, s) \mapsto \left(V_{\underline{t}}(\overline{x}), \qquad \begin{cases} q & \text{if } q \text{ is the smallest s.t. } q \in \underline{q} \cap F, \\ 0 & \text{if no such } q \text{ exists.} \end{cases} \right)$$

Now define $E \subseteq \mathbb{N}^{|Q|.|\delta|+1}$ to be such that $(\overline{v_1}, \ldots, \overline{v_{|Q|}}, q) \in E$ iff $\overline{v_q} \in C$; E is semilinear. We adjoin $\overline{0}$ to E iff $\overline{0} \in C$, in order to deal with the empty word. Now, by Assertion 16, a word w is accepted by the DetAPA $(\underline{A}, U, E)$ iff there exists a path in A from q_0 to $q \in F$, labeled by w, and whose Parikh image belongs to C, i.e., $w \in L(A, C)$.

Finally, recall that $L(\underline{A}) = L(A)$ and note that $\mathcal{M}(U)$ is finite as $\mathcal{M}(V)$ is: the extra component of U only adds a column and a row of 0's to the matrices. □

We note, for completeness, that constraint-deterministic CA, and thus finite-monoid DetAPA, strictly generalize DetPA. Indeed, the language $\{a,b\}^* \cdot \{a^n b^n \mid n \in \mathbb{N}^+\}$ is not expressible by a DetPA [5], but is expressible as a constraint-deterministic CA (A, C) where A is:



and C constrains the two loops on q_3 and q_4 to occur the same number of times. As any word in $\{a,b\}^*$ has at most one accepting path in A, this PA is constraint-deterministic.

5.2. *From finite-monoid DetAPA of bounded language to DetPA*

Let us first recall the following classical result:

Lemma 17 (e.g., [11]) *Let $u, v \in \Sigma^*$. Then $(u+v)^*$ is bounded iff there exists $z \in \Sigma^*$ such that $u, v \in z^*$.*

We will need the following technical lemma. Bounded languages being closed under morphism, for all automata A if $\mathsf{Run}(A)$ is bounded then so is $L(A)$. The converse is true when A is deterministic (and false otherwise):

Lemma 18. *Let A be a deterministic automaton for a bounded language, then* $\mathsf{Run}(A)$ *is bounded. Moreover,* $\mathsf{Run}(A)$ *is expressible as a SLRE whose branches* *are of the form $\rho_1\pi_1^* \cdots \rho_n\pi_n^*\rho_{n+1}$ where $\rho_i \neq \varepsilon$ for all $1 < i \leq n$ and the first transition of π_i differs from that of ρ_{i+1} for every i (including $i = n$ if $\rho_{n+1} \neq \varepsilon$).*

Proof. Recall that bounded languages are closed under *deterministic* rational transduction (see, e.g., [11]). Let a deterministic automaton $A = (Q, \Sigma, \delta, q_0, F)$ accept a bounded language and define the automaton A' as a copy of A over the alphabet $\Sigma \times \delta$ where a transition t is relabeled $(\mu(t), t)$. Then $\tau_{A'}$, the deterministic rational transduction defined by A', is such that $\mathsf{Run}(A) = \tau_{A'}(L(A))$, and thus $\mathsf{Run}(A)$ is bounded.

It will be useful to note the claim that if $X\pi_1^*\pi_2^*Y \subseteq \mathsf{Run}(A)$ for some nonempty paths π_1, π_2 and some bounded languages X and Y, then for some path π, $X\pi_1^*\pi_2^*Y \subseteq X\pi^*Y \subseteq \mathsf{Run}(A)$. To see this, note that if $X\pi_1^*\pi_2^*Y \subseteq \mathsf{Run}(A)$ then $X(\pi_1 + \pi_2)^*Y \subseteq \mathsf{Run}(A)$ because π_1 and π_2 are loops on a same state. Now $X(\pi_1 + \pi_2)^*Y$ is bounded because $\mathsf{Run}(A)$ is bounded, hence $(\pi_1 + \pi_2)^*$ is bounded. So pick π such that $\pi_1, \pi_2 \in \pi^*$ (by Lemma 17). Then $X(\pi_1 + \pi_2)^*Y \subseteq X\pi^*Y$. But π is a loop in A because $\pi_1 = \pi^j$ for some $j > 0$ is a loop so that $\mathsf{From}(\pi) = \mathsf{To}(\pi)$ in A. Hence $X\pi^*Y \subseteq \mathsf{Run}(A)$. Thus $X\pi_1^*\pi_2^*Y \subseteq X(\pi_1 + \pi_2)^*Y \subseteq X\pi^*Y \subseteq \mathsf{Run}(A)$.

Let E be a SLRE for $\mathsf{Run}(A)$, and consider one of its branches $P = \rho_1\pi_1^* \cdots \rho_n\pi_n^*\rho_{n+1}$. We assume n to be minimal among the set of all n' such that $\rho_1\pi_1^* \cdots \rho_n\pi_n^*\rho_{n+1} \subseteq \rho_1'\pi_1'^* \cdots \rho_{n'}'\pi_{n'}'^*\rho_{n'+1}' \subseteq \mathsf{Run}(A)$ for some $\rho_1', \pi_1', \ldots, \rho_{n'}', \pi_{n'}', \rho_{n'+1}'$.

First we do the following for $i = n, n-1, \ldots, 1$ in that order. If $\pi_i = \zeta\pi$ and $\rho_{i+1} = \zeta\rho$ for some maximal nonempty path ζ and for some paths π and ρ, we rewrite $\rho_i\pi_i^*\rho_{i+1}$ as $\rho_i'\pi_i'^*\rho_{i+1}'$ by letting $\rho_i' = (\rho_i\zeta)$, $\pi_i' = (\pi\zeta)$ and $\rho_{i+1}' = \rho$. This leaves the language of P unchanged and ensures at the ith stage that the first transition of π_j' (if any) differs from that of ρ_{j+1}' (if any) for $i \leq j \leq n$. Note that n has not changed.

Let $\rho_1'\pi_1'^* \cdots \rho_n'\pi_n'^*\rho_{n+1}'$ be the expression for P resulting from the above process. By the minimality of n, $\pi_i' \neq \varepsilon$ for $1 \leq i \leq n$. And for the same reason, $\rho_i' \neq \varepsilon$ for $1 < i \leq n$, since $\rho_i' = \varepsilon$ implies $X\pi_{i-1}'^*\pi_i'^*Y \subseteq Xz^*Y \subseteq \mathsf{Run}(A)$ for some z by the claim above, where $X = \rho_1' \cdots \pi_{i-2}'^*\rho_{i-1}'$ and $Y = \rho_{i+1}'\pi_{i+1}'^* \cdots \rho_{n+1}'$ are bounded languages. □

We are now ready to show the result of this section:

Lemma 19. *Let (A, U, C) be a finite-monoid DetAPA such that $L(A)$ is bounded. Then there exist a finite number of rflat DetCA having $L(A, U, C)$ as the union of their languages.*

Proof. Let $A = (Q_A, \Sigma, \delta, q_0, F_A)$ be a deterministic automaton whose language is bounded, let $U \colon \delta^* \to \mathcal{F}_d$ for some $d > 0$ be a morphism such that $\mathcal{M}(U)$ is finite, and let $C \subseteq \mathbb{N}^d$ be a semilinear set.

By the finiteness of $\mathcal{M}(U)$, every $M \in \mathcal{M}(U)$ has a minimum *threshold* and a strictly positive minimum *period* such that $M^{\text{threshold+period}} = M^{\text{threshold}}$. Let

$$0 < s = \max\{\text{threshold}(M) : M \in \mathcal{M}(U)\} + 1 \text{ and}$$
$$0 < p = \text{lcm}\{\text{period}(M) : M \in \mathcal{M}(U)\}.$$

Then every $M \in \mathcal{M}(U)$ verifies $M^{j+p} = M^j$ for every $j \geq s$.

Our main task will be to show the following:

Assertion 20. *For $n \geq 0$, $\rho_1, \pi_1, \ldots, \rho_n, \pi_n, \rho_{n+1} \in \delta^*$ satisfying the hypothesis of Lemma 18 and $s \leq j_1, \ldots, j_n < s + p$, there is an automaton D such that (A, C) is a rflat DetPA with:*

(1) *The initial state of D has no incoming transition and at most one outgoing transition, which is labeled by the first transition of ρ_1 if $\rho_1 \neq \varepsilon$,*
(2) *$\Psi(L(D))$ is $\rho_1 \pi_1^{j_1} (\pi_1^p)^* \rho_2 \pi_2^{j_2} (\pi_2^p)^* \cdots \rho_n \pi_n^{j_n} (\pi_n^p)^* \rho_{n+1}$,*
(3) *$\forall \omega \in L(D), \tilde{\Phi}(\omega) = U_{\Psi(\omega)}(\overline{0})$.*

We first show how Assertion 20 implies the result. Consider the set $\mathsf{Run}(A)$ of accepting paths in A; it is, by Lemma 18, a bounded language. Let P be the language defined a branch $\rho_1 \pi_1^* \cdots \rho_n \pi_n^* \rho_{n+1}$ of the SLRE for $\mathsf{Run}(A)$ given by Lemma 18. For $0 \leq j_1, \ldots, j_n < s + p$, define:

$$P_{(j_1, \ldots, j_n)} = \rho_1 \pi_1^{j_1} (\pi_1^p)^* \cdots \rho_n \pi_n^{j_n} (\pi_n^p)^* \rho_{n+1}.$$

Then P can be (redundantly) described as:

$$P = \bigcup_{0 \leq j_1, \ldots, j_n < s+p} P_{(j_1, \ldots, j_n)}.$$

Now, for some $0 \leq j_1, \ldots, j_n < s + p$, we argue that the language $\{\pi \in P_{(j_1, \ldots, j_n)} \mid U_\pi(\overline{0}) \in C\}$ is the union of the languages of some rflat DetPA. If each j_i is greater than s, then this is the statement of Assertion 20. Otherwise, if $P_{(j_1, \ldots, j_n)}$ is $\underbrace{\rho_1 \cdots}_{\alpha} \rho_i \pi_i^{j_i} (\pi_i^p)^* \rho_{i+1} \underbrace{\cdots \rho_{n+1}}_{\beta}$ with $j_i < s$, it can be expressed as $\alpha \rho_i \pi_i^{j_i + mp} (\pi_i^p)^* \rho_{i+1} \beta$ together with

$$\bigcup_{\ell=0}^{m-1} \alpha \rho_i \pi_i^{j_i + \ell p} \rho_{i+1} \beta, \tag{1}$$

where $m = \min\{\ell : j_i + \ell p \geq s\}$. Now $\alpha \rho_i \pi_i^{j_i + \ell p} \rho_{i+1} \beta$ can be rewritten as $\alpha \rho'_i \beta$, with the first transition of π_{i-1} still different from the first transition of $\rho'_i = \rho_i \pi_i^{j_i + \ell p} \rho_{i+1}$. Instances of $j_{i'} < s$ occurring in $\alpha \rho'_{i-1} \beta$ for $i' \neq i$ can be rewritten as well. When all occurrences of $j_i < s$ have been processed in this way, each resulting language is the language of a rflat DetPA by Assertion 20.

Now $\{\pi \in P \mid U_\pi(\overline{0}) \in C\}$ is thus the union of the languages of rflat DetPA (D, C). Now define D' as the automaton D where a label $(t, \overline{u})$ in D appears as $(\mu_A(t), \overline{u})$ in D', we argue that (D', C) is still a rflat DetPA: for any two transitions $(q, (a, \overline{u}), q')$ and $(q, (a, \overline{v}), q'')$ in D', there are two transitions $(q, (t, \overline{u}), q')$

and $(q,(t',\overline{v}),q'')$ with $\mu_A(t)=\mu_A(t')$ in D. Since t and t' may appear after the same prefix in $\Psi(L(D))$ (as any state of a rflat automaton is both accessible and co-accessible) and as $\Psi(L(D))$ is a set of paths, this implies that $\mathsf{From}(t)=\mathsf{From}(t')$. In turn, as A is deterministic, this implies that $t=t'$, and thus, as (D,C) is a DetPA, that $\overline{u}=\overline{v}$ and $q'=q''$, i.e., the two transitions we considered in D' are the same, hence (D',C) is a DetPA. Moreover, since D is rflat and D' has the same graph as D, D' is rflat. Finally, $L(D',C)=\mu(L(D,C))$, and thus $\{\mu(\pi)\mid \pi\in P\wedge U_\pi(\overline{0})\in C\}$ is the union of rflat DetPA languages. Going through all the branches thus leads to a finite set of rflat DetPA languages, with $L(A,U,C)$ as their union.

We now prove Assertion 20 by induction on n, the number of π_i's. For succinctness, we construct automata where the labels are pairs $(\pi,\overline{v})$ where $\pi=t_1t_2\cdots t_k$ is a nonempty word over δ; this is to be understood as a string of transitions with fresh states in between, with the first transition labeled $(t_1,\overline{v})$ and the other ones $(t_i,\overline{0})$, $i\geq 2$.

Suppose $n=0$, then we are only given ρ_1. If $\rho_1=\varepsilon$, then D is a single initial and final state. Otherwise, D is an automaton with states $\{q_0,q_f\}$, q_0 initial and q_f final, with a single transition, between q_0 and q_f, labeled $(\rho_1,U_{\rho_1}(\overline{0}))$. This verifies the conclusions of Assertion 20.

Suppose $n>0$. We introduce a few notations: for a path $\pi=t_1t_2\cdots t_k\in\delta^*$, we write M_π for $M_{t_k}\ldots M_{t_2}M_{t_1}$ and Δ_π for $U_\pi(\overline{0})$. Note that, using those notations, $U_\pi=(M_\pi,\Delta_\pi)$. Let $\pi\in\rho_1\pi_1^{j_1}(\pi_1^p)^*\rho_2\pi_2^{j_2}(\pi_2^p)^*\cdots\rho_n\pi_n^{j_n}(\pi_n^p)^*\rho_{n+1}$, and write $\pi=\rho_1\pi_1^{j_1}\pi_1^{pk}\gamma$ with k maximal. Define $M=M_{\rho_2\pi_2^{j_2}\ldots\rho_n\pi_n^{j_n}\rho_{n+1}}$, and note that $M_\gamma=M$, by the finite-monoid property. Then:

$$
\begin{aligned}
\Delta_\pi &= U_\gamma(U_{\pi_1^{j_1}}(U_{\pi_1^{pk}}(U_{\rho_1}(\overline{0}))))\\
&= M(M_{\pi_1^{j_1}}.U_{\pi_1^{pk}}(\Delta_{\rho_1})+\Delta_{\pi_1^{j_1}})+\Delta_\gamma && (\text{as } M_\gamma=M)\\
&= M.\Delta_{\pi_1^{j_1}}+M.M_{\pi_1^{j_1}}(M_{\pi_1^p}.U_{\pi_1^{p(k-1)}}(\Delta_{\rho_1})+\Delta_{\pi_1^p})+\Delta_\gamma\\
&= M.\Delta_{\pi_1^{j_1}}+M.M_{\pi_1^{j_1}}(U_{\pi_1^{p(k-1)}}(\Delta_{\rho_1})+\Delta_{\pi_1^p})+\Delta_\gamma && (\text{as } M_{\pi_1^{j_1}}.M_{\pi_1^p}=M_{\pi_1^{j_1}})\\
&\;\;\vdots && (\text{repeating the two previous lines})\\
&= M.\Delta_{\pi_1^{j_1}}+M.M_{\pi_1^{j_1}}(\Delta_{\rho_1}+k.\Delta_{\pi_1^p})+\Delta_\gamma\\
&= \underbrace{M.\Delta_{\pi_1^{j_1}}+M.M_{\pi_1^{j_1}}.\Delta_{\rho_1}}_{K}+k.\underbrace{M.M_{\pi_1^{j_1}}.\Delta_{\pi_1^p}}_{K'}+\Delta_\gamma.
\end{aligned}
$$

Note that the values of K and K' are independent of γ and k. Now construct D' as the automaton with states $\{q_0,q_f\}$, a transition between q_0 and q_f labeled $(\rho_1\pi_1^{j_1},K)$, and a transition from and to q_f labeled (π_1^p,K'). Let D'' be the automaton given by the induction hypothesis on $\rho_2,\pi_2,\ldots,\rho_n,\pi_n,\rho_{n+1},j_2,\ldots,j_n$. We construct D by merging D' and D'': we set q_0 initial and identify q_f with the initial state of D''. Note that (D,C) is indeed a flat DetPA: by induction hypothesis there is no cycle on the initial state of D'' and D'' is rflat, thus D is rflat; moreover, since D'' either has an empty language (if $\rho_2 = \varepsilon$, and thus $n = 1$) or starts with the first transition of ρ_2, which differs from that of π_1, (D, C) is a DetPA.

We argue that D fulfills the conclusions of Assertion 20. Point (1) is clear. Point (2) is verified thanks to the induction hypothesis: the projection of the language of D' is indeed $\rho_1\pi_1^{j_1}(\pi_1^p)^*$, and that of D'' is $\rho_2 \cdots \rho_{n+1}$. Finally, for (3): let $\omega \in L(D)$, and write $\omega = \omega_1\omega_2^k\omega_3$ with $\Psi(\omega_1) = \rho_1\pi_1^{j_1}$, $\Psi(\omega_2) = \pi_1^p$, and k maximal. Note that ω_3 is read over D''. Then:

$$\begin{aligned}\tilde{\Phi}(\omega) &= \tilde{\Phi}(\omega_1) + k.\tilde{\Phi}(\omega_2) + \tilde{\Phi}(\omega_3) = K + k.K' + \tilde{\Phi}(\omega_3)\\ &= K + k.K' + U_{\Psi(\omega_3)}(\overline{0}) \qquad\qquad \text{(by induction hypothesis)}\\ &= K + k.K' + \Delta_{\Psi(\omega_3)} = \Delta_{\Psi(\omega)} = U_{\Psi(\omega)}(\overline{0}).\end{aligned}$$

This concludes the proof of Assertion 20. □

We note that there exist bounded languages with nonsemilinear Parikh image in $\mathcal{L}_{\text{DetAPA}}$ (e.g., $\{a^nb^{2^n}\}$), thus there exist bounded languages in $\mathcal{L}_{\text{DetAPA}} \setminus \mathcal{L}_{\text{PA}}$.

5.3. *Proof of Theorem* 11 *and effectiveness*

Proof of Theorem 11. Let $Y \in \mathcal{L}_{\text{BoundedPA}}$. By Theorem 6, $Y \in \text{BSL}$, so let $w_1, \ldots, w_n$ be a socle of Y with $C = \mathsf{Iter}_{(w_1,\ldots,w_n)}(L)$ semilinear. Let (A, E) be the canonical ε-CA for $w_1, \ldots, w_n$ subject to C obtained by applying Construction 5. By construction, $L(A)$ is bounded. Moreover, (A, E) is constraint-deterministic: if two accepting paths π_1 and π_2 in A have the same label w, then π_1 and π_2 describe two ways to iterate the words in the socle of Y to get w. As the semilinear set C describes *all* ways to iterate these words to get a specific label, $\Phi(\pi_1) \in E$ iff $\Phi(\pi_2) \in E$.

Now applying Lemma 15 to the ε-CA (A, E) yields a finite-monoid DetAPA for Y whose underlying automaton has the bounded language $L(A)$. In turn, Lemma 19 yields a finite number of rflat DetCA having Y as the union of their languages. □

We note that all the constructions are effective, in the sense that given words $w_1, \ldots, w_n$ and a semilinear set $C \subseteq \mathbb{N}^n$, we can construct a DetCA for $\{w_1^{i_1} \cdots w_n^{i_n} \mid (i_1, \ldots, i_n) \in C\}$. We leave open whether there is an effective procedure to determinize a PA when the promise is made that its language is bounded.

6. Discussion and Further Work

We showed that PA and DetPA recognize the same class of bounded languages, namely BSL. To this end, we used related models (e.g., APA) and provided expressiveness results of independent interest (e.g., related to constraint-determinism and the finite-monoid property). Moreover, we noted that the union of rflat DetCA is a concept that has already been defined as 1-CQDD [4], showing that 1-CQDD capture exactly BSL. The closure properties observed in Corollaries 7 and 8 also apply to 1-CQDD, thus providing alternative proofs to those appearing in [4]. In particular, given $\mathcal{L}_{\text{1-CQDD}} = \text{BSL}$, the proof of the closure of 1-CQDD under concatenation is a consequence of the simple proofs of closure of $\mathcal{L}_{\text{PA}}$ and BOUNDED under concatenation, thus avoiding the long and technical proof of [4].

A related model, *reversal-bounded multi-counter machines* (*RBCM*) [14], has been shown to have the same expressive power as PA [17]. It is known that *one-way* deterministic RBCM are strictly more powerful than DetPA [5], thus our result carries over to RBCM, showing that RBCM and one-way deterministic RBCM recognize the same class of bounded languages, namely BSL. This provides an alternative proof of the same fact appearing in a recent paper of Ibarra and Seki [15], and yields as a by-product a characterization of BSL using a model provably weaker than one-way deterministic RBCM.

Further work includes an in-depth study of the finite-monoid property of APA. In particular, we suspect that finite-monoid APA are as expressive as PA, and that finite-monoid DetAPA are as expressive as constraint-deterministic CA. One further avenue of research is to investigate the related decision problems, e.g., is it decidable whether the language of a PA is bounded? or whether it is that of a constraint-deterministic CA?

Acknowledgments

The first author thanks Laurent Beaudou, Marc Kaplan, and Adrien Lemaître for stimulating discussions and comments on early versions of this paper. We further thank the anonymous referees for their critical comments and especially for pointing out a flawed claim made in an earlier version of this work, namely that finite-monoid DetAPA are equivalent to DetPA.

References

[1] Rajeev Alur & P. Madhusudan (2009): *Adding nesting structure to words*. *J. ACM* 56, pp. 16:1–16:43, doi:10.1145/1516512.1516518.

[2] Sébastien Bardin, Alain Finkel, Jérôme Leroux & Philippe Schnoebelen (2005): *Flat acceleration in symbolic model checking*. In: *Proceedings of the 3rd International Symposium on Automated Technology for Verification and Analysis (ATVA'05)*, *LNCS* 3707, Springer, Taipei, Taiwan, pp. 474–488, doi:10.1007/11562948_35.

[3] Mikolaj Bojanczyk (2009): *Weak MSO with the Unbounding Quantifier*. In: *STACS 2009*, 3, pp. 159–170, doi:10.4230/LIPIcs.STACS.2009.1834.

[4] Ahmed Bouajjani & Peter Habermehl (1999): *Symbolic reachability analysis of FIFO-channel systems with nonregular sets of configurations*. *Theoretical Computer Science* 221(1–2), pp. 211–250, doi:10.1016/S0304-3975(99)00033-X.

[5] Michaël Cadilhac, Alain Finkel & Pierre McKenzie (2011): *On the Expressiveness of Parikh Automata and Related Models*. In Rudolf Freund, Markus Holzer, Carlo Mereghetti, Friedrich Otto & Beatrice Palano, editors: *NCMA*, *books@ocg.at* 282, Austrian Computer Society, pp. 103–119.

[6] Christian Choffrut, Andreas Malcher, Carlo Mereghetti & Beatrice Palano (2010): *On the Expressive Power of FO*[+]. In Adrian-Horia Dediu, Henning Fernau & Carlos Martn-Vide, editors: *Language and Automata Theory and Applications*, *Lecture Notes in Computer Science* 6031, Springer Berlin/Heidelberg, pp. 190–201, doi:10.1007/978-3-642-13089-2_16.

[7] Flavio D'Alessandro & Stefano Varricchio (2008): *On the growth of context-free languages*. *J. Autom. Lang. Comb.* 13, pp. 95–104.

[8] Stéphane Demri, Alain Finkel, Valentin Goranko & Govert van Drimmelen (2010): *Model-checking* (*CTL**) *over Flat Presburger Counter Systems*. *Journal of Applied Non-Classical Logics* 20(4), pp. 313–344, doi:10.3166/jancl.20.313-344.

[9] Alain Finkel, S. Purushothaman Iyer & Grégoire Sutre (2003): *Well-abstracted transition systems: application to FIFO automata*. *Information and Computation* 181(1), pp. 1–31, doi:10.1016/S0890-5401(02)00027-5.

[10] Patrick C. Fischer (1965): *Multi-tape and infinite-state automata — A survey*. *Commun. ACM* 8(12), pp. 799–805, doi:10.1145/365691.365962.

[11] Seymour Ginsburg (1966): *The Mathematical Theory of Context-Free Languages*. McGraw-Hill, Inc., New York, NY, USA.

[12] Seymour Ginsburg & Edwin H. Spanier (1964): *Bounded ALGOL-Like Languages*. *Transactions of the American Mathematical Society* 113(2), pp. 333–368, doi:10.2307/1994067.

[13] Seymour Ginsburg & Edwin H. Spanier (1966): *Bounded Regular Sets*. *Proceedings of the American Mathematical Society* 17(5), pp. 1043–1049, doi:10.2307/2036087.

[14] Oscar H. Ibarra (1978): *Reversal-Bounded Multicounter Machines and Their Decision Problems*. *J. ACM* 25(1), pp. 116–133, doi:10.1145/322047.322058.

[15] Oscar H. Ibarra & Shinnosuke Seki (2011): *Characterizations of Bounded Semilinear Languages by One-Way and Two-way Deterministic Machines*. In: *Automata and Formal Languages*, pp. 211–224.

[16] Michael Kaminski & Nissim Francez (1994): *Finite-memory automata*. *TCS* 134(2), pp. 329–363, doi:10.1016/0304-3975(94)90242-9.

[17] Felix Klaedtke & Harald Rueß (2003): *Monadic Second-Order Logics with Cardinalities*. In: *International Colloquium on Automata, Languages and Programming*, *LNCS* 2719, Springer-Verlag, pp. 681–696, doi:10.1007/3-540-45061-0_54.

[18] Nils Klarlund & Fred B. Schneider (1989): *Verifying Safety Properties Using Nondeterministic Infinite-state Automata*. Technical Report, Ithaca, NY, USA.

[19] Rohit J. Parikh (1966): *On context-free languages*. *Journal of the ACM* 13(4), pp. 570–581, doi:10.1145/321356.321364.

[20] Howard Straubing (1994): *Finite Automata, Formal Logic, and Circuit Complexity*. Birkhäuser, Boston, doi:10.1007/978-1-4612-0289-9.