Context-Freeness of Word-MIX Languages

Ryoma Sin'ya

Akita University
ryoma@math.akita-u.ac.jp**

Abstract. In this paper we provide a decidable characterisation of the context-freeness of a Word-MIX language $L_A(w_1, \ldots, w_k)$, where $L_A(w_1, \ldots, w_k)$ is the set of all words over A that contain the same number of subword occurrences of parameter words $w_1, \ldots, w_k$.

1 Introduction

Counting occurrences of letters in words is a major topic in formal language theory. In particular, much ink has been spent on investigating the counting ability of some language classes. For example, Joshi et al. [1] suggested that the language MIX $= \{w \in \{a, b, c\}^* \mid |w|_a = |w|_b = |w|_c\}$ should not be in the class of so-called mildly context-sensitive languages since it allows too much freedom in word order, so that relations between MIX and several language classes have been investigated (*e.g.*, indexed languages [2], range concatenation languages [3], tree-adjoining languages [4], multiple context-free languages [5], *etc.*). The Parikh map is another rich example on this topic (counting occurrences of letters) [6].

In the recent work [7] by Colbourn et. al., the counting feature of MIX is generalised from counting *letter* occurrences to counting *word* occurrences. They considered several problems for languages of the form $L_A(w_1, \ldots, w_k) = \{w \in A^* \mid |w|_{w_1} = \cdots = |w|_{w_k}\}$ (where $|u|_v$ is the number of occurrences of v in w) which we call *Word-MIX languages* (WMIX for short) in this paper. While $L_A(w_1, w_2)$ is always deterministic context-free, it can also be regular ($L_A(ab, ba)$ is regular if $A = \{a, b\}$, while it is not regular if $A = \{a, b, c\}$, for example) [7]. This kind of generalisation – from letter occurrences to word occurrences – is also considered in the context of the Parikh map through so-called Parikh matrices [8] and subword histories [9, 10] (in this setting they have considered *scattered* subword occurrences instead of subword occurrences).

Colbourn et. al. [7] provided a necessary and sufficient condition for w_1 and w_2 for the WMIX language $L_A(w_1, w_2)$ to be regular, and gave a polynomial time algorithm for testing that condition. For the fully general case, the decidability of the regularity problem for WMIX languages can be derived from some known results on *unambiguous constrained automata* (UnCA for short), since $L_A(w_1, \ldots, w_k)$ is always recognised by an UnCA, and the regularity for UnCA languages is decidable due to [11].

** The author is also with RIKEN AIP.

In this paper, we show that *context-freeness is decidable* for WMIX languages. We also give an alternative decidability proof for the regularity of WMIX languages. As we mentioned above, the regularity for WMIX languages is already known to be decidable thanks to the decidability results on UnCA languages (which include all WMIX languages) given by Cadilhac et al [11]. But the alternative proof of the regularity for WMIX languages given in this paper gives more *structural information* of WMIX languages, and the proof can be naturally extended into the context-freeness. We introduce a new notion called *dimension*, which represents certain structural information of WMIX languages, and prove that a WMIX language is (1) regular if and only if its dimension is at most one, and (2) context-free if and only if its dimension is at most two. To the best of our knowledge, there has been no research on the context-freeness for WMIX languages or UnCA languages. As far as we know, a language class with such a decidable context-freeness property is very rare. We are only aware of such examples in some subclasses of bounded languages [12–14] and languages associated with vector addition systems [15].

2 Preliminaries

For a set X, we denote by $\#(X)$ the cardinality of X. We denote by $\mathbb{N}$ the set of natural numbers including 0. We call a mapping $M : X \to \mathbb{N}$ multiset over X. For a set X, we write 2^X for the power set of X.

We assume that the reader has a basic understanding of automata and linear algebra.

2.1 Words and Word-MIX languages

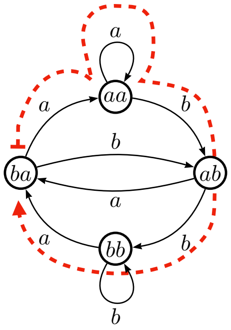
For an alphabet A, we denote the set of all words (resp. all non-empty words) over A by A^* (resp. A^+). We write A^n (resp. $A^{<n}$) for the set of all words of length n (resp. less than n), and write $\mathbb{N}^{\leq c}$ for the set of all natural numbers less than or equal c for $c \in \mathbb{N}$. For a pair of words $v, w \in A^*$, $|w|_v$ denotes the number of subword occurrences of v in w

$$|w|_v \overset{\texttt{def}}{=} \#(\{(w_1, w_2) \in A^* \times A^* \mid w_1 v w_2 = w\})\,.$$

We write $u \sqsubseteq v$ if u is a subword of v, and write $u \sqsubseteq_{\mathrm{sc}} v$ if u is a scattered subword of v. For words $w_1, \ldots, w_k \in A^*$, we define

$$L_A(w_1, \ldots, w_k) \overset{\texttt{def}}{=} \{w \in A^* \mid |w|_{w_1} = \cdots = |w|_{w_k}\}$$

and call it the *Word-MIX* (WMIX for short) *language of k-parameter words* $w_1, \ldots, w_k$ *over* A. For a word $w \in A^*$, we denote the set of prefixes and suffixes of w by $\mathrm{pref}(w)$ and $\mathrm{suff}(w)$, and denote the length-n ($n \leq |w|$) prefix and suffix of w by $\mathrm{pref}_n(w)$ and $\mathrm{suff}_n(w)$, respectively.



$$\mathtt{walk}_{\mathcal{G}_A^2}(ba, aabba)$$
$$= (ba, aa, aa, ab, bb, ba).$$

$$\mathtt{word}_{\mathcal{G}_A^2}((ba, aa, aa, ab, bb, ba))$$
$$= baaabba.$$

Fig. 1. The 2-dimensional de Bruijn graph $\mathcal{G}_A^2$ over $A = \{a, b\}$, a walk (ba, aa, aa, ab, bb, ba) (dotted red arrow) on $\mathcal{G}_A^2$ and its corresponding word $baaabba$.

2.2 Graphs and Walks

Let $\mathcal{G} = (V, E)$ be a (directed) graph. We call a sequence of vertices $\omega = (v_1, \ldots, v_n) \in V^n\,(n \geq 1)$ *walk* (from v_1 into v_n in $\mathcal{G}$) if $(v_i, v_{i+1}) \in E$ for each $i \in \{1, \ldots, n-1\}$, and define the length of ω as $n-1$ and denote it by $|\omega|$. We denote by $\mathtt{from}(\omega)$ and $\mathtt{into}(\omega)$ the source $\mathtt{from}(\omega) \overset{\mathsf{def}}{=} v_1$ and the target $\mathtt{into}(\omega) \overset{\mathsf{def}}{=} v_n$ of ω. ω is called an *empty walk* if $|\omega| = 0$. If two walks $\omega_1 = (v_1, \ldots, v_m), \omega_2 = (v'_1, \ldots, v'_n)$ are connectable (*i.e.*, $\mathtt{into}(\omega_1) = \mathtt{from}(\omega_2)$), we write $\omega_1 \odot \omega_2$ for the connecting walk $\omega_1 \odot \omega_2 \overset{\mathsf{def}}{=} (v_1, \ldots, v_m, v'_2, \ldots, v'_n)$. A non-empty walk ω is called *loop* (on $\mathtt{from}(\omega)$) if $\mathtt{from}(\omega) = \mathtt{into}(\omega)$. A walk $(v_1, \ldots, v_n)$ is called *path* if $v_i \neq v_j$ for every $i, j \in \{1, \ldots, n\}$ with $i \neq j$. A loop $(v, v_1, \ldots, v_n, v)$ is called *cycle* if $(v, v_1, \ldots, v_n)$ is a path. We use the metavariable π for a path, and the metavariable γ for a cycle. For a cycle γ and $n \geq 1$, we write γ^n for the loop which is an n-times repetition of γ. We denote by $\mathcal{W}(\mathcal{G}), \mathcal{P}(\mathcal{G})$, and by $\mathcal{C}(\mathcal{G})$ the set of all walks, paths and cycles in $\mathcal{G}$. Note that $\mathcal{W}(\mathcal{G})$ is infinite in general, but $\mathcal{P}(\mathcal{G})$ and $\mathcal{C}(\mathcal{G})$ are both finite if $\mathcal{G}$ is finite.

The N*-dimensional de Bruijn graph* $\mathcal{G}_A^N = (A^N, E)$ over A is a graph whose vertex set A^N is the set of words of length N and the edge set E is defined by

$$E \overset{\mathsf{def}}{=} \{(av, vb) \mid a, b \in A, v \in A^{N-1}\}.$$

The case $N = 2$ is depicted in Fig. 1.

Let v be a vertex of $\mathcal{G}_A^N$. A word $w = a_1 \cdots a_m \in A^+$ induces the walk $(v, v_1, \ldots, v_m)$ (where $v_i = \mathrm{suff}_n(v\,\mathrm{pref}_i(w))$) in $\mathcal{G}_A^N$, and we denote it by $\mathtt{walk}_{\mathcal{G}_A^N}(v, w)$. Conversely, a walk $\omega = (v_1, \ldots, v_n)$ in $\mathcal{G}_A^N$ induces the word $v_1 \mathrm{suff}_1(v_2) \cdots \mathrm{suff}_1(v_n) \in A^*$, and we denote it by $\mathtt{word}_{\mathcal{G}_A^N}(\omega)$ (see Fig. 1). For words $w, w_1, \ldots, w_k \in A^*$ and a walk $\omega = (v_0, v_1, \ldots, v_n) \in \mathcal{W}(\mathcal{G}_A^N)$, we define the following vectors in $\mathbb{N}^k$:

$$|w|_{(w_1, \ldots, w_k)} \overset{\mathsf{def}}{=} (|w|_{w_1}, \ldots, |w|_{w_k})$$

$$|\omega|_{(w_1, \ldots, w_k)} \overset{\mathsf{def}}{=} \sum_{i=1}^{n} (c_{i,1}, \ldots, c_{i,k}) \text{ where } c_{i,j} = 1 \text{ if } w_j \in \mathrm{suff}(v_i), c_{i,j} = 0 \text{ otherwise.}$$

We call $|w|_{(w_1,\ldots,w_k)}$ (resp. $|\omega|_{(w_1,\ldots,w_k)}$) the *occurrence vector of w (resp. ω)*. We notice that the range of the summation in the above definition of $|\omega|_{(w_1,\ldots,w_k)}$ *does not contain* 0, hence $|\omega|_{(w_1,\ldots,w_k)} = (0,\ldots,0)$ if ω is an empty walk $\omega = (v_0)$. The next proposition states a basic property of $\mathcal{G}_A^N$, which can be shown by a straightforward induction on the length of w.

Proposition 1. *Let $w_1,\ldots,w_k \in A^*$ and $N = \max(|w_1|,\ldots,|w_k|)$. For any pair of words $v, w \in A^*$ such that $|v| = N$ and $\omega = \mathtt{walk}_{\mathcal{G}_A^N}(v,w)$, we have*

$$|vw|_{(w_1,\ldots,w_k)} = |v|_{(w_1,\ldots,w_k)} + |\omega|_{(w_1,\ldots,w_k)}.$$

2.3 Well-Quasi-Orders

A quasi order $\le$ on a set X is called *well-quasi-order* (*wqo* for short) if any infinite sequence $(x_i)_{i\in\mathbb{N}}$ $(x_i \in X)$ contains an increasing pair $x_i \le x_j$ with $i < j$. Let $\le_1$ be a quasi order on a set X_1 and $\le_2$ be a quasi order on a set X_2. The *product order* $\le_{1,2}$ is a quasi order on $X_1 \times X_2$ defined by

$$(x_1, y_1) \le_{1,2} (x_2, y_2) \overset{\mathtt{def}}{\iff} x_1 \le_1 x_2 \text{ and } y_1 \le_2 y_2.$$

Proposition 2 **(*cf.* Proposition 6.1.1 in [16]).** *Let $\le_1$ be a wqo on a set X_1 and $\le_2$ be a wqo on a set X_2. The product order $\le_{1,2}$ is again a wqo on $X_1 \times X_2$.*

We list some examples of wqos below:

(1) The identity relation $=$ on any finite set X is a wqo (*the pigeonhole principle*).
(2) The usual order $\le$ on $\mathbb{N}$ is a wqo.
(3) The product order $\le_m$ on $\mathbb{N}^m$ is a wqo for any $m \ge 1$ (*Dickson's lemma*), which is a direct corollary of Proposition 2.
(4) The point-wise order $\le_{\mathtt{pt}}$ on the multisets $\mathbb{N}^X$ ($M \le_{\mathtt{pt}} M' \overset{\mathtt{def}}{\iff} M(x) \le M'(x)$ for all $x \in X$) over a finite set X is a wqo (just a paraphrase of Dickson's lemma).

3 Path-Cycle Decomposition of Walks

In this section, we provide a simple method which decomposes, in left-to-right manner, a walk ω into a (possibly empty) path π and a sequence of cycles Γ (Fig. 2). This decomposition, and its inverse operation (composition), are probably folklore, and the contents in this section appeared already in the author's unpublished note [17]. A similar method is also used in [11].

Let $\mathcal{G} = (V, E)$ be a graph. For a pair of sequences of cycles $\Gamma_1 = (\gamma_1,\ldots,\gamma_n)$, $\Gamma_2 = (\gamma'_1,\ldots,\gamma'_m)$, we write $\Gamma_1.\Gamma_2$ for the concatenation $(\gamma_1,\ldots,\gamma_n,\gamma'_1,\ldots,\gamma'_m)$. When $\Gamma_1 = (\gamma)$ we simply write $\gamma.\Gamma_2$ for $\Gamma_1.\Gamma_2$ We write $\emptyset$ for the empty sequence of cycles. For $\Gamma = (\gamma_1,\ldots,\gamma_n)$, we denote by $\Gamma(i)$ for the i-th component γ_i of Γ, and denote by $|\Gamma|_\gamma$ the number $\#(\{i \mid \Gamma(i) = \gamma\})$ of occurrences of γ in Γ. For a walk $\omega = (v_1,\ldots,v_n)$, we denote by $V(\omega)$ the set of all vertices appearing in ω: $V(\omega) \overset{\mathtt{def}}{=} \{v_1,\ldots,v_n\}$.

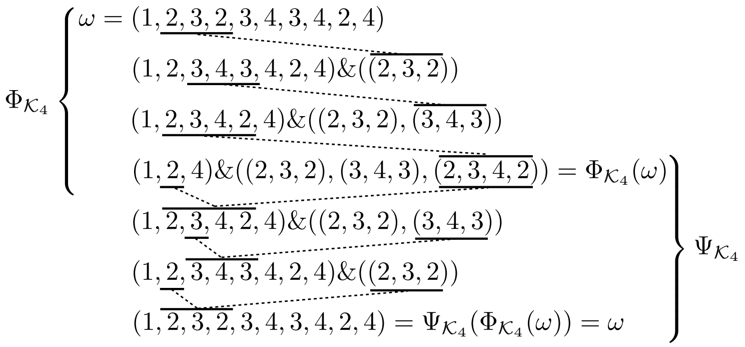

$\Phi_{\mathcal{K}_4}$
$\omega = (1, \underline{2, 3, 2,} 3, 4, 3, 4, 2, 4)$
$(1, 2, \underline{3, 4, 3,} 4, 2, 4)\&(\overline{(2, 3, 2)})$
$(1, \underline{2, 3, 4, 2,} 4)\&((2, 3, 2), \overline{(3, 4, 3)})$
$(1, \underline{2,} 4)\&((2, 3, 2), (3, 4, 3), \overline{(2, 3, 4, 2)}) = \Phi_{\mathcal{K}_4}(\omega)$
$(1, \overline{2, \underline{3}, 4, 2,} 4)\&((2, 3, 2), \underline{(3, 4, 3)})$
$(1, \underline{2,} \overline{3, 4, 3,} 4, 2, 4)\&(\underline{(2, 3, 2)})$
$(1, \overline{2, 3, 2,} 3, 4, 3, 4, 2, 4) = \Psi_{\mathcal{K}_4}(\Phi_{\mathcal{K}_4}(\omega)) = \omega$
$\Psi_{\mathcal{K}_4}$

Fig. 2. Computation of $\varPhi_{\mathcal{K}_4}$ and $\varPsi_{\mathcal{K}_4}$

We then define a decomposition function $\varPhi_{\mathcal{G}}$ inductively as follows: $\varPhi_{\mathcal{G}}((v)) \overset{\texttt{def}}{=} ((v), \emptyset)$ and

$$\varPhi_{\mathcal{G}}(\omega \odot (v, v')) \overset{\texttt{def}}{=} \begin{cases} (\pi \odot (v, v'),\ \varGamma) & \text{if } v' \notin V(\pi), \\ (\pi_1,\ \varGamma.(\pi_2 \odot (v, v'))) & \text{if } \pi = \pi_1 \odot (v') \odot \pi_2 \end{cases}$$
$$\text{where } (\pi, \varGamma) = \varPhi_{\mathcal{G}}(\omega).$$

It is clear by definition that, for any ω and $(\omega', \varGamma) = \varPhi_{\mathcal{G}}(\omega)$, ω' is a path and $\varGamma$ is a sequence of cycles, *i.e.*, $\varPhi_{\mathcal{G}} : \mathcal{W}(\mathcal{G}) \to \mathcal{P}(\mathcal{G}) \times \mathcal{C}(\mathcal{G})^*$. Conversely, we define a composition (partial) function $\varPsi_{\mathcal{G}} : \mathcal{W}(\mathcal{G}) \times \mathcal{C}(\mathcal{G})^* \rightharpoonup \mathcal{W}(\mathcal{G})$ inductively as follows: $\varPsi_{\mathcal{G}}(\omega, \emptyset) \overset{\texttt{def}}{=} \omega$ and

$$\varPsi_{\mathcal{G}}(\omega, \gamma.\varGamma) \overset{\texttt{def}}{=} \begin{cases} \pi \odot \gamma \odot \omega' & \text{if } \pi \odot (v) \odot \omega' = \varPsi_{\mathcal{G}}(\omega, \varGamma) \text{ where } \texttt{from}(\gamma) = v, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

The following propositions can be shown by a straightforward induction on the length of ω.

Proposition 3. *Let $\mathcal{G} = (V, E)$ be a graph and $\omega \in \mathcal{W}(\mathcal{G})$. Then the following hold for $(\pi, \varGamma) = \varPhi_{\mathcal{G}}(\omega)$:*

(1) $|\omega| = |\pi| + \sum_{i=1}^{|\varGamma|} |\varGamma(i)|$;
(2) $\omega = \varPsi_{\mathcal{G}}(\pi, \varGamma)$, *i.e.*, $\varPsi_{\mathcal{G}} \circ \varPhi_{\mathcal{G}}$ *is the identity function on* $\mathcal{W}(\mathcal{G})$.

Proposition 4. *Let $w_1, \ldots, w_k \in A^*$ and $N = \max(|w_1|, \ldots, |w_k|)$. For any ω in $\mathcal{W}(\mathcal{G}_A^N)$,*

$$|\omega|_{(w_1,\ldots,w_k)} = |\pi|_{(w_1,\ldots,w_k)} + \sum_{i=1}^{|\varGamma|} |\varGamma(i)|_{(w_1,\ldots,w_k)}$$

holds where $(\pi, \varGamma) = \varPhi_{\mathcal{G}_A^N}(\omega)$.

$$\begin{tikzcd}
abbbabbb & & \{(ab,bb),(bb,bb),(ab,bb,ba,ab)\} \\
\text{word } w \arrow[d, shift left, "\texttt{walk}_{\mathcal{G}_A^N}"] & (ab,bb)\,\&\,((bb,bb),(ab,bb,ba,ab),(bb,bb)) & \text{trace } T \\
\text{walk } \omega \arrow[u, shift left, "\texttt{word}_{\mathcal{G}_A^N}"] \arrow[r, shift left, "\Phi_{\mathcal{G}_A^N}"] & \text{pair } (\pi,\Gamma) \arrow[l, shift left, "\Psi_{\mathcal{G}_A^N}"] \arrow[r, "\mathbb{N}\mathrm{Tr}"] & \text{multi-trace } M \arrow[u, "\mathrm{Tr}"'] \\
(ab,bb,bb,ba,ab,bb,bb) & & (ab,bb)\mapsto 1,(bb,bb)\mapsto 2 \\
 & & (ab,bb,ba,ab)\mapsto 1
\end{tikzcd}$$

Fig. 3. Relations between words, walks and (multi-)traces ($N = 2$ for the examples).

Example 1. Consider the complete graph $\mathcal{K}_4 = (V_4 = \{1,2,3,4\}, E_4 = V_4 \times V_4)$ of order 4 and a walk $\omega = (1,2,3,2,3,4,3,4,2,4)$. The result of decomposition is $\Phi_{\mathcal{K}_4}(\omega) = (\pi = (1,2,4), \Gamma = ((2,3,2),(3,4,3),(2,3,4,2)))$. All intermediate computation steps of $\Phi_{\mathcal{K}_4}(\omega)$ and $\Psi_{\mathcal{K}_4}(\Phi_{\mathcal{K}_4}(\omega))$ are drawn in Fig. 2 (in the figure we denote by $\pi\&\Gamma$ a pair (π, Γ) for visibility). It is clear that all conditions in Proposition 3 are satisfied ($|\omega| = 9 = 2+2+2+3 = |\omega| + \sum_{i=1}^{3} |\Gamma(i)|$).

3.1 Multi-Traces and Traces

For a walk ω in a graph $\mathcal{G}$, we define the *multi-trace* $\mathbb{N}\mathrm{Tr}(\omega) : \mathcal{P}(\mathcal{G}) \cup \mathcal{C}(\mathcal{G}) \to \mathbb{N}$ of a walk ω as the following multiset over paths and cycles:

$$(\mathbb{N}\mathrm{Tr}(\omega))(\pi) \stackrel{\mathtt{def}}{=} \begin{cases} 1 & \text{if } \pi = \pi_\omega \\ 0 & \text{otherwise} \end{cases} \qquad (\mathbb{N}\mathrm{Tr}(\omega))(\gamma) \stackrel{\mathtt{def}}{=} |\Gamma|_\gamma$$
$$\text{where } (\pi_\omega, \Gamma) = \Phi_{\mathcal{G}}(\omega).$$

We define the *trace* $\mathrm{Tr}(\omega)$ of a walk ω in $\mathcal{G}$ as the following set of paths and cycles:

$$\mathrm{Tr}(\omega) \stackrel{\mathtt{def}}{=} \{\pi \in \mathcal{P}(\mathcal{G}) \mid (\mathbb{N}\mathrm{Tr}(\omega))(\pi) \neq 0\} \cup \{\gamma \in \mathcal{C}(\mathcal{G}) \mid (\mathbb{N}\mathrm{Tr}(\omega))(\gamma) \neq 0\}.$$

Intuitively, the multi-trace of ω in $\mathcal{G}$ is obtained by forgetting the ordering of the decomposition result $(\omega, \Gamma) = \Phi_{\mathcal{G}}(\omega)$ of ω, and the trace of ω is obtained by forgetting the multiplicity from the original multi-trace (see Fig. 3 for the relation).

Since Condition (1) in Proposition 3 and Proposition 4 do not depend on the order of a sequence Γ, one can easily observe that the following proposition holds by the definition of $\mathbb{N}\mathrm{Tr}(\omega)$.

Proposition 5. *Let $w_1, \ldots, w_k \in A^*$ and $N = \max(|w_1|, \ldots, |w_k|)$. For any ω in $\mathcal{W}(\mathcal{G}_A^N)$, we have*

$$|\omega|_{(w_1,\ldots,w_k)} = \sum_{\pi \in \mathcal{P}(\mathcal{G}_A^N)} (\mathbb{N}\mathrm{Tr}(\omega))(\pi) \cdot |\pi|_{(w_1,\ldots,w_k)} + \sum_{\gamma \in \mathcal{C}(\mathcal{G}_A^N)} (\mathbb{N}\mathrm{Tr}(\omega))(\gamma) \cdot |\gamma|_{(w_1,\ldots,w_k)}.$$

For a set $T \subseteq \mathcal{P}(\mathcal{G}) \cup \mathcal{C}(\mathcal{G})$, the following proposition states that we can effectively test whether T is a trace or not.

Proposition 6. *Let $T \subseteq \mathcal{P}(\mathcal{G}) \cup \mathcal{C}(\mathcal{G})$ be a set of paths and cycles in a graph $\mathcal{G} = (V, E)$. The following are equivalent:*

(1) *T is a trace of some walk in $\mathcal{G}$.*
(2) *T can be written as $T = \{\pi\} \cup \{\gamma_1, \ldots, \gamma_m\}$ such that (i) $V(\gamma_m) \cap V(\pi) = \{\mathtt{from}(\gamma_m)\}$ and (ii) for every $i \in \{1, \ldots, m-1\}$, $V(\pi') \cap V(\gamma_i) = \{\mathtt{from}(\gamma_i)\}$ where $\pi' \odot (\mathtt{from}(\gamma_i)) \odot \omega = \Psi_{\mathcal{G}}(\pi, (\gamma_{i+1}, \ldots, \gamma_m))$.*

4 Main Results

In this section we first introduce a new notion for WMIX languages called *dimension*. Afterwards, we state our main results that characterise both regularity and context-freeness of WMIX languages.

Definition 1. Let $w_1, \ldots, w_k \in A^*$ and $N = \max(|w_1|, \ldots, |w_k|)$. Let $T = \{\pi\} \cup \{\gamma_1, \ldots, \gamma_m\}$ be a trace of a walk ω in $\mathcal{G}_A^N$. A subset S of $\{\gamma_1, \ldots, \gamma_m\}$ is called *pumpable in T of $L_A(w_1, \ldots, w_k)$* if, for any number $n \geq 1$, there exists a word $uv \in L_A(w_1, \ldots, w_k)$ with $\omega = \mathtt{walk}_{\mathcal{G}_A^N}(u, v)$ such that (1) $\mathrm{Tr}(\omega) = T$ and (2) $(\mathbb{N}\mathrm{Tr}(\omega))(\gamma) \geq n$ for each $\gamma \in S$. We further say S is *maximal* if no proper superset of S included in $\{\gamma_1, \ldots, \gamma_m\}$ is pumpable.

Remark 1. The emptyset $\emptyset$ is always pumpable in a trace T of $L_A(w_1, \ldots, w_k)$ such that $\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v)) = T$ for some $uv \in L_A(w_1, \ldots, w_k)$. Moreover, it is decidable whether S is pumpable or not in T of $L_A(w_1, \ldots, w_k)$ because the condition in the above definition can be reduced to a formula in *Presburger arithmetic* as follows: Let $S = \{\gamma_{i_1}, \ldots, \gamma_{i_j}\}$ and $u = \mathtt{from}(\pi)$. One can easily observe that S is pumpable if and only if the following first-order formula

$$\begin{aligned} \varphi_S \overset{\mathtt{def}}{=} & \forall n \in \mathbb{N}\ \exists x_1, \ldots, x_m \in \mathbb{N} \\ & \Big(\forall i \in \{1, \ldots, m\}\,(x_i > 0) \wedge \forall i \in \{i_1, \ldots, i_j\}\,(x_{i_j} \geq n) \\ & \quad \wedge \mathtt{diff}\Big(|u|_{(w_1,\ldots,w_k)} + |\pi|_{(w_1,\ldots,w_k)} + \sum_{i=1}^{m} x_i \cdot |\gamma_i|_{(w_1,\ldots,w_k)}\Big) = 0\Big) \end{aligned}$$

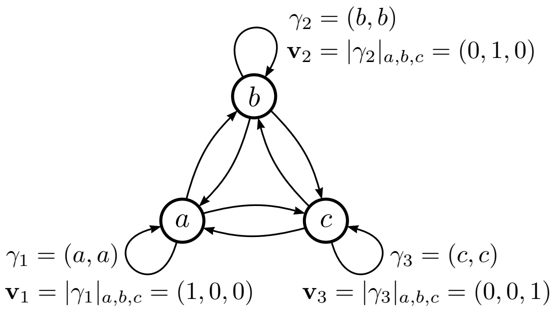
is true. Thanks to the decidability of the validity of Presburger arithmetic, we can algorithmically check the validity of φ_S, *i.e.*, the pumpability of S.

Recall that a *vector space* is a set $\boldsymbol{V} \subseteq \mathbb{R}^k$ such that $\mathbf{0} \in \boldsymbol{V}, \boldsymbol{V} + \boldsymbol{V} \subseteq \boldsymbol{V}$ and $\mathbb{R}\boldsymbol{V} = \{\alpha \cdot \boldsymbol{v} \mid \boldsymbol{v} \in \boldsymbol{V}, \alpha \in \mathbb{R}\} \subseteq \boldsymbol{V}$ where $\mathbf{0}$ is the vector with all zeros.

Definition 2. Let $w_1, \ldots, w_k \in A^*$, $N = \max(|w_1|, \ldots, |w_k|)$. The *dimension* of $L = L_A(w_1, \ldots, w_k)$ is the natural number defined as

$$\max\{\dim(\boldsymbol{V}) \mid \boldsymbol{V} = \mathrm{span}(\{|\gamma|_{(w_1,\ldots,w_k)} \mid \gamma \in S\}), S \text{ is pumpable in some } T \text{ of } L\}$$

where $\dim(\boldsymbol{V})$ is the dimension of the vector space $\boldsymbol{V}$ and $\mathrm{span}(B)$ is the vector space spanned by B (where $\mathrm{span}(\emptyset) \overset{\mathtt{def}}{=} \{\mathbf{0}\}$).



$\gamma_2 = (b, b)$
$\mathbf{v}_2 = |\gamma_2|_{a,b,c} = (0, 1, 0)$
b
a
c
$\gamma_1 = (a, a)$
$\mathbf{v}_1 = |\gamma_1|_{a,b,c} = (1, 0, 0)$
$\gamma_3 = (c, c)$
$\mathbf{v}_3 = |\gamma_3|_{a,b,c} = (0, 0, 1)$

Fig. 4. The 1-dimensional de Bruijn graph $\mathcal{G}_A^1$ over $A = \{a, b, c\}$.

The dimension of a WMIX language L is, roughly speaking, the minimum number of cycles (in the de Bruijn graph) that should be counted *independently*. We describe this intuition more rigorously by using MIX $= L_A(a, b, c)$ for $A = \{a, b, c\}$ as a simple example.

Example 2. Since $\max(|a|, |b|, |c|) = 1$, it is enough to consider the 1-dimensional de Bruijn graph $\mathcal{G}_A^1$ over $A = \{a, b, c\}$ (see Fig. 4). One can easily observe that the set of cycles $S = \{\gamma_1 = (a, a), \gamma_2 = (b, b), \gamma_3 = (c, c)\}$, each γ_i is depicted in Fig. 4, is pumpable in the trace $T = \{(a, b, c)\} \cup S$: for any $n > 0$, the word $aw_n = a^{n+1}b^{n+1}c^{n+1}$ is in MIX and it satisfies the two conditions in the Definition 1 as (1) $\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(a, w_n)) = T$ and (2) $(\mathbb{N}\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(a, w_n)))(\gamma_i) = n$ for each $\gamma_i \in S$. The occurrence vectors corresponding to $\gamma_1, \gamma_2, \gamma_3$ are $\boldsymbol{v_1} = (1, 0, 0), \boldsymbol{v_2} = (0, 1, 0), \boldsymbol{v_3} = (0, 0, 1)$, respectively. Since those occurrence vectors are linearly independent, the vector space spanned by them is $\mathbb{R}^3$ and thus the dimension of MIX is three.

By considering dimensions of WMIX languages, we can nicely characterise both regularity and context-freeness as follows.

Theorem 1 (regularity). *$L_A(w_1, \ldots, w_k)$ is regular if and only if its dimension is at most one.*

Theorem 2 (context-freeness). *$L_A(w_1, \ldots, w_k)$ is context-free if and only if its dimension is at most two.*

Some pushdown automaton $\mathcal{A}$ can recognise $L_A(a, b)$ since, by using its stack, $\mathcal{A}$ can track the number $|w|_a - |w|_b$. However, no pushdown automaton $\mathcal{A}$ can recognise MIX $= L_A(a, b, c)$ since, for that purpose, one should track the numbers $|w|_a - |w|_b$ and $|w|_b - |w|_c$ simultaneously. This is a rough intuition why a language with dimension greater than or equal three is never to be context-free (the formal proof is in the next section).

The set $\mathcal{P}(\mathcal{G}_A^N) \cup \mathcal{C}(\mathcal{G}_A^N)$ of paths and cycles in the N-dimensional de Bruijn graph is finite, hence we can effectively enumerate all traces of all walks in $\mathcal{G}_A^N$ thanks to Proposition 6. Moreover, as we mentioned in Remark 1, we can also effectively enumerate all pumpable sets in a trace. For a pumpable set S, computing the dimension of the vector space spanned by the occurrence vectors S is just counting the maximum number of linearly independent ones from the occurrence vectors of S. Combining these facts and Theorem 1–2, we can effectively compute the dimension of $L_A(w_1, \ldots, w_k)$ and hence we have the following decidability result.

Corollary 1. *Regularity and context-freeness are decidable for WMIX languages.*

5 Proof of the Main Results

The proof structure of Theorem 1 is similar with one of Theorem 2, albeit that the latter is more complicated. In this section, we firstly investigate some structural properties of pumpable sets, which play crucial role in the main proof. We secondly give a proof of Theorem 1 which would give a good intuition for the latter proof. Finally, we give a proof of Theorem 2.

5.1 Properties of Pumpable Sets

For a vector $\boldsymbol{v} = (c_1, \ldots, c_k) \in \mathbb{R}^k$, we define $\texttt{diff}(\boldsymbol{v}) \overset{\texttt{def}}{=} \sum_{i=1}^{k}(\max\{c_1, \ldots, c_k\} - c_i)$. Observe that $w \in L_A(w_1, \ldots, w_k)$ if and only if $\texttt{diff}(|w|_{(w_1,\ldots,w_k)}) = 0$.

Lemma 1. *Let $w_1, \ldots, w_k \in A^*$ and $N = \max(|w_1|, \ldots, |w_k|)$. For any maximum pumpable set S in $T = \{\pi\} \cup S'$ of $L_A(w_1, \ldots, w_k)$, if $\boldsymbol{V} = \mathrm{span}(\{|\gamma|_{(w_1,\ldots,w_k)} \mid \gamma \in S\})$ has a non-zero dimension, then $\boldsymbol{V}$ contains the vector $\mathbf{1}$.*

Proof. Let $u = \texttt{from}(\pi)$. Since S is pumpable, there exists an infinite sequence $(uv_i)_{i \in \mathbb{N}}$, where $u \in A^N$, of words that satisfies:

(1) $\mathrm{Tr}(\texttt{walk}_{\mathcal{G}_A^N}(u, v_i)) = T$ for all $i \in \mathbb{N}$.
(2) $uv_i \in L_A(w_1, \ldots, w_k)$ for all $i \in \mathbb{N}$.
(3) $\mathbb{N}\mathrm{Tr}(\texttt{walk}_{\mathcal{G}_A^N}(u, v_i))(\gamma) < \mathbb{N}\mathrm{Tr}(\texttt{walk}_{\mathcal{G}_A^N}(u, v_j))(\gamma)$ for all $i, j \in \mathbb{N}$ with $i < j$ and for all $\gamma \in S$.

Now consider an infinite sequence of multi-traces of the above sequence

$$(M_i)_{i \in \mathbb{N}} \overset{\texttt{def}}{=} (\mathbb{N}\mathrm{Tr}(\texttt{walk}_{\mathcal{G}_A^N}(u, v_i)))_{i \in \mathbb{N}}.$$

Since the point-wise order on the multisets over any finite set is a wqo (thanks to Dickson's lemma) and $\mathcal{P}(\mathcal{G}_A^N) \cup \mathcal{C}(\mathcal{G}_A^N)$ is finite, $(M_i)_{i \in \mathbb{N}}$ contains an infinite increasing subsequence $(M_j)_{j \in J}$ $(J \subseteq \mathbb{N})$. Let $\overline{S} = (S' \setminus S)$. Because S is maximum, the number of maximum occurrence of any non-pumpable cycle $\gamma \in \overline{S}$ is bounded, *i.e.*, there is some constant $c \in \mathbb{N}$ such that $(\mathbb{N}\mathrm{Tr}(\texttt{walk}_{\mathcal{G}_A^N}(u, v_i)))(\gamma) < c$ for any $\gamma \in \overline{S}$ and $i \in \mathbb{N}$. By using pigeonhole principle, we can deduce that, in the infinite sequence $(M_j)_{j \in J}$, there exists a pair $(i_1, i_2) \in J^2$ with $i_1 < i_2$ such that $(M_{i_1})(\gamma) = (M_{i_2})(\gamma)$ for all $\gamma \in \overline{S}$. Let $C = \sum_{\gamma \in \overline{S}} M_{i_1}(\gamma) \cdot |\gamma|_{(w_1,\ldots,w_k)}$. Combining the above observation and the condition (3) of $(uv_i)_{i\in\mathbb{N}}$, we have

$$M_{i_1}(\gamma) = M_{i_2}(\gamma) \text{ for all } \gamma \in \overline{S} \qquad M_{i_1}(\gamma) < M_{i_2}(\gamma) \text{ for all } \gamma \in S. \qquad (\bigstar)$$

Because $uv_{i_1}, uv_{i_2} \in L_A(w_1, \ldots, w_k)$, by Proposition 1 and Proposition 5, we have

$$\begin{aligned} &\mathtt{diff}(|uv_{i_1}|_{(w_1,\ldots,w_k)}) = \mathtt{diff}(|uv_{i_2}|_{(w_1,\ldots,w_k)}) = 0 \\ =&\mathtt{diff}\left(|u|_{(w_1,\ldots,w_k)} + |\pi|_{(w_1,\ldots,w_k)} + C + \sum_{\gamma\in S} M_{i_1}(\gamma)\cdot|\gamma|_{(w_1,\ldots,w_k)}\right) \\ =&\mathtt{diff}\left(|u|_{(w_1,\ldots,w_k)} + |\pi|_{(w_1,\ldots,w_k)} + C + \sum_{\gamma\in S} M_{i_2}(\gamma)\cdot|\gamma|_{(w_1,\ldots,w_k)}\right). \end{aligned}$$

Moreover, from the above equation we obtain

$$\mathtt{diff}\left(\sum_{\gamma\in S}(M_{i_2}(\gamma) - M_{i_1}(\gamma))\cdot|\gamma|_{(w_1,\ldots,w_k)}\right) = 0 \qquad (1)$$

because for any $\boldsymbol{v}$ such that $\mathtt{diff}(\boldsymbol{v}) = 0$, $\mathtt{diff}(\boldsymbol{v} + \boldsymbol{v'}) = 0$ if and only if $\mathtt{diff}(\boldsymbol{v'}) = 0$. By Condition ($\bigstar$), the vector

$$\boldsymbol{v} = \sum_{\gamma\in S}(M_{i_2}(\gamma) - M_{i_1}(\gamma))\cdot|\gamma|_{(w_1,\ldots,w_k)}$$

is not the zero vector $\mathbf{0}$. Thus $\boldsymbol{v}$ is of the form $n \cdot \mathbf{1}$ ($n \neq 0$), *i.e.*, $\mathbf{1} \in \boldsymbol{V}$. □

Lemma 2. *Let $w_1, \ldots, w_k \in A^*$ and $N = \max(|w_1|, \ldots, |w_k|)$. For any trace T of some walk in $\mathcal{G}_A^N$, if $\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u,v)) = T$ for some $uv \in L_A(w_1, \ldots, w_k)$, then there exists a unique maximal (i.e., the maximum) pumpable set S in T of $L_A(w_1, \ldots, w_k)$.*

Proof. Let S_1, S_2 be two maximal pumpable sets in T of $L_A(w_1, \ldots, w_k)$ and $S_1 = \{\gamma_1, \ldots, \gamma_m\}$. We now prove that $S_1 \cup S_2$ is also pumpable in T, which implies $S_1 = S_2$ by the maximality of S_1 and S_2. By Condition ($\bigstar$) and Equation (1) in the proof of Lemma 1, we can deduce that there exist $n_1, \ldots, n_m \in \mathbb{N}$ such that $n_i > 0$ for all $i \in \{1, \ldots, m\}$ and $\mathtt{diff}(\sum_{i=1}^{m} n_i \cdot |\gamma_i|_{(w_1,\ldots,w_k)}) = 0$. Let $(uv_i)_{i\in\mathbb{N}}$ be an infinite sequence that ensures the pumpability of S_2, namely,

(1) $\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v_i)) = T$ for all $i \in \mathbb{N}$.
(2) $uv_i \in L_A(w_1, \ldots, w_k)$ for all $i \in \mathbb{N}$.
(3) $\mathbb{N}\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v_i))(\gamma) \geq i$ for all $i \in \mathbb{N}$ and for all $\gamma \in S_2$.

Let uv'_i be a word that satisfying $\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v'_i)) = T$, $\mathbb{N}\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v'_i))(\gamma_j) = \mathbb{N}\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v_i))(\gamma_j) + i \times n_j$ for all $i \in \mathbb{N}$ and $\gamma_j \in S_1$. Such word uv'_i always exists because we can just pump an occurrence of $\gamma_j \in S_1$ in $\mathtt{walk}_{\mathcal{G}_A^N}(u, v_i)$

$(i \times n_j)$-times repeatedly. Then the infinite sequence $(uv'_i)_{i \in \mathbb{N}}$ satisfies $uv'_i \in L_A(w_1, \ldots, w_k)$ and $\mathbb{N}\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v'_i))(\gamma) \geq i$ for all $i \in \mathbb{N}$ and for all $\gamma \in S_1 \cup S_2$, because $\mathtt{diff}(\sum_{j=1}^{m} n_j \cdot |\gamma_j|_{(w_1,\ldots,w_k)}) = 0$. Which means that $(uv'_i)_{i \in \mathbb{N}}$ ensures the pumpability of $S_1 \cup S_2$, this ends the proof. □

Lemma 3. *Let $w_1, \ldots, w_k \in A^*$ and $N = \max(|w_1|, \ldots, |w_k|)$. For any maximum pumpable set S of $L_A(w_1, \ldots, w_k)$,*

(1) *if the vector space $\boldsymbol{V}$ spanned by the occurrence vectors of S is of dimension one, then $\boldsymbol{V} = \mathrm{span}(\{\mathbf{1}\})$ where $\mathbf{1}$ is the k-dimensional vector with entries all 1, i.e., any occurrence vector $\boldsymbol{v}$ of S satisfies $\mathtt{diff}(\boldsymbol{v}) = 0$.*
(2) *if the vector space $\boldsymbol{V}$ spanned by the occurrence vectors of S is of dimension greater than or equal two, then we can choose a basis $B \subseteq \{|\gamma|_{(w_1,\ldots,w_k)} \mid \gamma \in S\}$ of $\boldsymbol{V}$ such that any element $\boldsymbol{v}$ of B satisfies $\mathtt{diff}(\boldsymbol{v}) \neq 0$.*

Proof. Condition (1) is a direct consequence of Lemma 1. Condition (2) is also from Lemma 1. Let $\gamma \in S$ be a pumpable cycle such that $\mathtt{diff}(\gamma) \neq 0$. Such γ always exists since S contains at least two cycles whose occurrence vectors are linearly independent. Moreover, by Condition (★) in the proof of Lemma 1, we can deduce that there exists $B' \subseteq S$ such that the occurrence vectors of $B' \cup \{\gamma\}$ are linearly independent and $\mathbf{1} \in \mathrm{span}(B' \cup \{\gamma\})$. Thus any vector of the form $n \cdot \mathbf{1}\,(n \neq 0)$ is not in the occurrence vectors of $B' \cup \{\gamma\}$, we can take a desired basis B as an extension of $B' \cup \{\gamma\}$ $(B' \cup \{\gamma\} \subseteq B)$. □

5.2 Proof of Theorem 1

To prove "only if" part, we modify standard Pumping Lemma as follows and call it Shrinking Lemma.

Lemma 4 (Shrinking Lemma for regular languages). *Let $L \subseteq A^*$ be a regular language. Then there exists a constant $c \in \mathbb{N}$ such that, for any number $n \geq c$ and for any word $w \in L$ with $|w| \geq n$, for any factorisation $w = xyz$ such that $|y| = n \geq c$, there exists a word y' such that (1) $y' \sqsubseteq_{\mathrm{sc}} y$, (2) $|y'| \leq c$ and (3) $xy'z \in L$.*

Proof. Let $\mathcal{A} = (Q, \delta, q_0, F)$ be a deterministic automaton recognising L. We show that the constant $c = \#(Q) - 1$ satisfies the condition stated in the above lemma. Let $n \geq c$, $w \in L$ with $|w| \geq n$ and $w = xyz$ be a factorisation such that $|y| = n$. Since w is recognised by $\mathcal{A}$, we have an accepting computation

$$q_0 \xrightarrow[\mathcal{A}]{x} p \xrightarrow[\mathcal{A}]{y} q \xrightarrow[\mathcal{A}]{z} q' \in F$$

for some states $p, q \in Q$ and $q' \in F$. Remark that $p \xrightarrow[\mathcal{A}]{y} q$ means that the automaton $\mathcal{A}$ reads y and moves from state p to q. Obviously, by using similar technique in previous path-cycle decomposition of walks, we can obtain a scattered subword y' of y such that $p \xrightarrow[\mathcal{A}]{y'} q$ (thus $xy'z \in L$) and $|y'| \leq c$. □

Now we prove Theorem 1. Let $N = \max(|w_1|, \ldots, |w_k|)$. The "only if" part is shown by contraposition. Assume that the dimension of $L = L_A(w_1, \ldots, w_k)$ is two (higher-dimensional case can be shown similarly). Because L is of dimension two, there exists a maximum pumpable set $S = \{\gamma_{i_1}, \ldots, \gamma_{i_j}\}$ in some trace $T = \{\pi\} \cup \{\gamma_1, \ldots, \gamma_m\}$ in $\mathcal{G}_A^N$ such that two occurrence vectors $|\gamma_\alpha|_{(w_1,\ldots,w_k)}$ and $|\gamma_\beta|_{(w_1,\ldots,w_k)}$ of two cycles γ_α and γ_β in S are linearly independent and any occurrence vector of an element of S can be represented as a linear combination of $|\gamma_\alpha|_{(w_1,\ldots,w_k)}$ and $|\gamma_\beta|_{(w_1,\ldots,w_k)}$. By Condition (2) of Lemma 3, we can assume that $\mathtt{diff}(|\gamma_\alpha|_{(w_1,\ldots,w_k)}) \neq 0$ and $\mathtt{diff}(|\gamma_\beta|_{(w_1,\ldots,w_k)}) \neq 0$. Since S is a maximum pumpable set and the dimension of L is two, there exists a constant $c_T \in \mathbb{N}$ such that for any $n \in \mathbb{N}$ there exists a word $uv_n \in L$ with $\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v_n)) = T$, $(\mathbb{N}\mathrm{Tr}(u, v_n))(\gamma_\alpha) = n_\alpha$, $(\mathbb{N}\mathrm{Tr}(u, v))(\gamma_\beta) = n_\beta \geq n$ and $(\mathbb{N}\mathrm{Tr}(u, v_n))(\gamma_i) \leq c_T$ for each $i \in (\{1, \ldots, m\} \setminus \{\alpha, \beta\})$. By Proposition 5, we can assume that the walk $\mathtt{walk}_{\mathcal{G}_A^N}(u, v_n)$ is of the form

$$\mathtt{walk}_{\mathcal{G}_A^N}(u, v_n) = \omega_1 \odot \gamma_\alpha^{n_\alpha} \odot \omega_2 \odot \gamma_\beta^{n_\beta} \odot \omega_3.$$

Intuitively, $\mathtt{walk}_{\mathcal{G}_A^N}(u, v_n)$ firstly moves to $\mathtt{from}(\gamma_\alpha)$ (part of ω_1), and secondly passes γ_α repeatedly n_α-times and moves to $\mathtt{from}(\gamma_\beta)$ (part of $\gamma_\alpha^{n_\alpha} \odot \omega_2$), and lastly passes γ_β repeatedly n_β-times and moves to the end (part of $\gamma_\beta^{n_\beta} \odot \omega_3$). If L is regular, then by Lemma 4, there exists a constant c such that for any $n \geq c$ and the factorisation $uv_n = xy_nz_n$, where x, y_n and z_n are words corresponding to the first, second and last part of walks described above, there exists a word y'_n satisfying conditions (1)–(3) in Lemma 4. Because $\mathtt{diff}(|\gamma_\beta|_{(w_1,\ldots,w_k)}) \neq 0$, we have $|\gamma_\beta|_{w_j} < |\gamma_\beta|_{w_{j'}}$ for some $1 \leq j, j' \leq k$. However, since the length of x and y'_n are fixed by constant but z_n can be arbitrarily large, the gap of the occurrences $|z_n|_{w_{j'}} - |z_n|_{w_j}$ can be arbitrarily large (thus $|xy'_nz_n|_{w'_j} - |xy'_nz_n|_{w_j}$ can be arbitrarily large, too). It means that $xy'_nz_n \notin L$ for sufficiently large n, a contradiction.

The "if" part is achieved by showing that the language $L_T = \{uv \in L \mid |u| = N, \mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v)) = T\}$ is regular for each trace $T = \{\pi\} \cup \{\gamma_1, \ldots, \gamma_m\}$ in $\mathcal{G}_A^N$. It implies that L is regular because $L = L^{<N} \cup \bigcup_{T:\text{trace}} L_T$ (notice that $L^{<N} = \{w \in L \mid |w| < N\}$ is finite and thus regular). One can observe that $L = \{w \in L \mid |w| < N\} \cup \bigcup_{T:\text{ trace in } \mathcal{G}_A^N} L_T$, hence if every L_T is regular then L is also regular. To achieve it, we construct a deterministic automaton $\mathcal{A}_{T,S}$, where S is the maximum pumpable set in T, so that $L_T = L(\mathcal{A}_{T,S})$. Let $\overline{S} = (T \backslash S \backslash \{\pi\})$ and define

$$c_T \overset{\mathtt{def}}{=} \max\{(\mathbb{N}\mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v)))(\gamma) \in \mathbb{N} \mid uv \in L, \mathrm{Tr}(\mathtt{walk}_{\mathcal{G}_A^N}(u, v)) = T, \gamma \in \overline{S}\}$$

(notice that $\max \emptyset \overset{\mathtt{def}}{=} 0$ as usual). c_T is well-defined natural number, because, by the definition of pumpable set and S being maximum, for any cycle γ in T but not in S, the maximum number of occurrences of γ in a walk of some word in L is bounded. We denote by $\mathcal{F}$ the set of all functions from $\overline{S}$ to $\mathbb{N}^{\leq c_T}$. Notice that both $\overline{S}$ and $\mathbb{N}^{\leq c_T}$ are finite, $\mathcal{F}$ is also finite. Let $f_0 \in \mathcal{F}$ be the constant

$$\begin{aligned}
&Q \overset{\mathtt{def}}{=} A^{<N} \cup \{q_{\mathrm{rej}}\} \cup Q' \text{ where } Q' = (\mathcal{P}(\mathcal{G}_A^N) \times 2^S \times \mathcal{F})\\
&\delta \overset{\mathtt{def}}{=} \{(u, a, ua) \mid u \in A^{<(N-1)}, a \in A\} \cup \{(u, a, ((ua), \emptyset, f_0)) \mid u \in A^{(N-1)}, a \in A\}\\
&\qquad \cup \{(q_{\mathtt{rej}}, a, q_{\mathtt{rej}}) \mid a \in A\}\\
&\text{①} \quad \cup \big\{\big((\pi_1 \odot (aw), P, f), b, (\pi_1 \odot (aw, wb), P, f)\big) \mid a, b \in A, wb \notin V(\pi_1)\big\}\\
&\text{②} \quad \cup \big\{\big((\pi_1 \odot (wb) \odot \pi_2 \odot (aw), P, f), b, (\pi_1, P', f)\big) \mid a, b \in A, \pi_2 \odot (aw, wb) \in S,\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad P' = P \cup \{\pi_2 \odot (aw, wb)\}\big\}\\
&\text{③} \quad \cup \big\{\big((\pi_1 \odot (wb) \odot \pi_2 \odot (aw), P, f), b, q_{\mathrm{rej}}\big) \mid a, b \in A, \pi_2 \odot (aw, wb) \notin T\}\\
&\text{④} \quad \cup \big\{\big((\pi_1 \odot (wb) \odot \pi_2 \odot (aw), P, f), b, (\pi_1, P, f')\big) \mid a, b \in A, \gamma = \pi_2 \odot (aw, wb) \in \overline{S},\\
&\qquad\qquad f(\gamma) < c_T, f'(\gamma) = f(\gamma) + 1, f'(\gamma') = f(\gamma') \text{ for all } \gamma' \in (\overline{S} \setminus \{\gamma\})\}\\
&\text{⑤} \quad \cup \{\big((\pi_1 \odot (wb) \odot \pi_2 \odot (aw), P, f), b, q_{\mathrm{rej}}\big) \mid a, b \in A, \pi_2 \odot (aw, wb) \in \overline{S},\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad f(\pi_2 \odot (aw, wb)) \geq c_T\big\}\\
&F \overset{\mathtt{def}}{=} \{(\pi, S, f) \in Q' \mid f(\gamma) \geq 1 \text{ for all } \gamma \in \overline{S}, \mathtt{diff}(\boldsymbol{v}(\pi, f)) = 0\}\\
&\qquad \text{where } \boldsymbol{v}(\pi, f) = |\mathtt{from}(\pi)|_{(w_1,\ldots,w_k)} + |\pi|_{(w_1,\ldots,w_k)} + \sum_{\gamma \in \overline{S}} f(\gamma) \cdot |\gamma|_{(w_1,\ldots,w_k)}
\end{aligned}$$

Fig. 5. The construction of $\mathcal{A}_{T,S} = (Q, \delta, \varepsilon, F)$.

map to 0. Then the construction is as follows: $\mathcal{A}_{T,S} = (Q, \delta, \varepsilon, F)$ where each component is defined in Fig. 5.

Although the formal definition in Fig. 5 could look complex, the behavior of $\mathcal{A}_{T,S}$ is simple: it computes path-cycle decomposition and counts the number of occurrences of each non-pumpable cycle $\gamma \in \overline{S}$. The main part of states is Q' which consists of the path part $\mathcal{P}(\mathcal{G}_A^N)$, pumpable-cycles part 2^S and non-pumpable-cycles part $\mathcal{F}$. While reading an input word w, $\mathcal{A}_{T,S}$ extends the path part (Case ①) if the next vertex wb is not in the current path. If the next vertex wb is already in the current path, there are four possibilities (Case ②–⑤). If the induced cycle γ on wb is in S (Case ②), $\mathcal{A}_{T,S}$ updates the pumpable cycle part. The number of occurrences of such cycle $\gamma \in S$ is not necessary to be memorised, since by Condition (1) of Lemma 3 $\mathtt{diff}(|\gamma|_{(w_1,\ldots,w_k)}) = 0$. If γ is not in T (Case ③), $\mathcal{A}_{T,S}$ goes to the rejecting state $q_{\mathtt{rej}}$, since the trace of w is never to be T. If γ is in $\overline{S}$, there are two possibilities further: if the current number of occurrences of γ is less than c_T (Case ④), $\mathcal{A}_{T,S}$ increments it, otherwise (Case ⑤), $\mathcal{A}_{T,S}$ goes to $q_{\mathtt{rej}}$ because w is never to be in L by the definition of c_T. □

5.3 Proof of Theorem 2

The proof structure is similar with the regular case (Theorem 1). The following lemma is a context-free variant of Lemma 4.

Lemma 5 (Shrinking Lemma for context-free languages). *Let $L \subseteq A^*$ be a context-free language. Then there exists a constant $c \in \mathbb{N}$ such that, for any* *number $n \geq c$ and for any word $w \in L$ with $|w| \geq n$, there exists a factorisation $w = xyz$ and a word y' such that (0) $2n > |y| \geq n \geq c$, (1) $y' \sqsubseteq_{\mathrm{sc}} y$, (2) $|y'| \leq c$ and (3) $xy'z \in L$.*

To prove this lemma, we need to prepare some notion and notation. Let $G = (V, D, X_0)$ be a context-free grammar over an alphabet A where V ($V \cap A = \emptyset$) is a finite set of *non-terminals*, $D \subseteq V \times (V \cup A \cup \{\epsilon\})^+$ is a finite set of *derivation rules*, and $X_0 \in V$. The set of (V, A)*-trees*, ranged over by T, is given by the following grammar:

$$T ::= a \ \ (a \in A \cup \{\epsilon\}) \mid X(T_1, \cdots, T_n) \ \ (X \in V, n \geq 1)$$

Namely, (V, A)-trees are trees whose internal nodes are non-terminals, and whose leaves are letters in A or the special symbol $\epsilon \notin A$. For a (V, A)-tree T, we denote by $NT)$ the set of all non-terminals appeared in T, and denote by $\texttt{root}\,(T)$ the root of T. The *yield* $\texttt{yield}(\cdot)$ is a function from (V, A)-trees into A^* defined inductively as $\texttt{yield}(a) = a, \texttt{yield}(\epsilon) = \varepsilon$ where ε is the empty string, and $\texttt{yield}(X(T_1, \ldots, T_n)) = \texttt{yield}(T_1) \cdots \texttt{yield}(T_n)$. We call a $(V, A \cup \{[\,]\})$-tree C *context* if exactly one leaf of C is the special symbol $[\,] \notin A$.We denote by $C[T]$ the (V, A)-tree obtained by replacing $[\,]$ in C by T. We define *the set $\mathcal{T}(G)$ of derivation trees of G* as

$$\begin{aligned}\mathcal{T}(G) \overset{\texttt{def}}{=} \{T : (V, A)\text{-tree} \mid \texttt{root}\,(T) = X_0,\ \text{for each context } C \\ T = C[X(T_1, \ldots, T_n)] \text{ implies } (X, \texttt{root}\,(T_1) \cdots \texttt{root}\,(T_n)) \in D\}\end{aligned}$$

and define $\mathcal{L}(G) \overset{\texttt{def}}{=} \{\texttt{yield}(T) \mid T \in \mathcal{T}(G)\}$.

We call a (V, A)-tree T *simple* if, for any path in T from the root to a leaf, no non-terminal appears more than once.

Proof (of Lemma 5). Let $G = (V, D, X_0)$ be a context-free grammar over an alphabet A. We assume that G is in Chomsky normal form (*i.e.*, any derivation tree $T \in \mathcal{T}(G)$ is unary-binary tree and no leaf is ϵ unless $T = X_0(\epsilon)$) without loss of generality. We show that the constant

$$c \overset{\texttt{def}}{=} \max\{|\texttt{yield}(S)\,| \mid S \text{ is simple and } S \sqsubseteq_{\mathrm{sc}} T \text{ for some } T \in \mathcal{T}(G)\}$$

satisfies the condition stated in the lemma above (since the set of all simple (V, A)-trees is finite, the above constant is well-defined). Let $n \geq c$, $w \in L$ with $|w| \geq n$ and $T \in \mathcal{T}(G)$ with $\texttt{yield}(T) = w$. One can easily prove that, by a straightforward induction on the size of trees, for any unary-binary tree with at least n leaves contains a subtree with the number of leaves at least n and less than $2n$. Since $|w| \geq n$, T has at least n leaves and thus there exists a subtree T' such that $c \leq n \leq |\texttt{yield}(T')\,| < 2n$ and $T = C[T']$ for some C. Then we can "shrink" T' into S, by applying the decomposition method presented in [18, 19], so that S is a simple scattered subtree of T' and $C[S]$ is also in $\mathcal{T}(G)$. Then for the factorisation $w = xyz$ where $x[\,]z = \texttt{yield}(C)$ and $y = \texttt{yield}(T')$ satisfies the condition (0), and $y' = \texttt{yield}(S)$ satisfies the condition (1)–(2) since S is simple, $S \sqsubseteq_{\mathrm{sc}} T'$ and $C[S] \in \mathcal{T}(G)$. □

Now we prove Theorem 2. Let $N = \max(|w_1|, \ldots, |w_k|)$. The "only if" part is shown by contraposition. Assume that the dimension of $L = L_A(w_1, \ldots, w_k)$ is three (higher-dimensional case can be shown similarly). Because L is of dimension three, there exists a maximum pumpable set $S = \{\gamma_{i_1}, \ldots, \gamma_{i_j}\}$ in some trace $T = \{\pi\} \cup \{\gamma_1, \ldots, \gamma_m\}$ in $\mathcal{G}_A^N$ such that three occurrence vectors $B = \{|\gamma_\alpha|_{(w_1,\ldots,w_k)}, |\gamma_\beta|_{(w_1,\ldots,w_k)}, |\gamma_\delta|_{(w_1,\ldots,w_k)}\}$ of three cycles γ_α, γ_β and γ_δ in S are linearly independent and any occurrence vector of an element of S can be represented as a linear combination of B. By Condition (2) of Lemma 3, we can assume that any vector $\boldsymbol{v}$ in B satisfies $\texttt{diff}(\boldsymbol{v}) \neq 0$. Since S is a maximum pumpable set and the dimension of L is three, there exists a constant $c_T \in \mathbb{N}$ such that for any $n \in \mathbb{N}$ there exists a word $uv_n \in L$ with $\mathrm{Tr}(\texttt{walk}_{\mathcal{G}_A^N}(u, v_n)) = T$, $(\mathbb{N}\mathrm{Tr}(\texttt{walk}_{\mathcal{G}_A^N}(u, v_n)))(\gamma_i) = n_i \geq n$ for each $i \in \{\alpha, \beta, \delta\}$ and $(\mathbb{N}\mathrm{Tr}(\texttt{walk}_{\mathcal{G}_A^N}(u, v_n)))(\gamma_i) \leq c$ for each $i \in (\{1, \ldots, m\} \setminus \{\alpha, \beta, \delta\})$. By Proposition 5, we can assume that the walk $\texttt{walk}_{\mathcal{G}_A^N}(u, v_n)$ is of the form

$$\texttt{walk}_{\mathcal{G}_A^N}(u, v_n) = \omega_1 \odot \gamma_\alpha^{n_\alpha} \odot \omega_2 \odot \gamma_\beta^{n_\beta} \odot \omega_3 \odot \gamma_\delta^{n_\delta} \odot \omega_4.$$

Let $u_{n,1}, u_{n,2}$ and $u_{n,3}$ be words corresponding to $\omega_1 \odot \gamma_\alpha^{n_\alpha}, \omega_2 \odot \gamma_\beta^{n_\beta}$ and $\omega_3 \odot \gamma_\delta^{n_\delta} \odot \omega_4$, respectively (thus $uv_n = u_{n,1}u_{n,2}u_{n,3}$). Let $M_n = \min\{n_\alpha \cdot |\gamma_\alpha|, n_\beta \cdot |\gamma_\beta|, n_\delta \cdot |\gamma_\delta|\}$. If L is context-free, then by Lemma 5, there exists a constant c such that for any $n \geq c$, there is a factorisation $uv_n = x_n y_n z_n$ and a word y'_n satisfying conditions (0)–(3) in Lemma 5. Take $n \in \mathbb{N}$ that satisfies $M_n \geq c$. Then, the word y in the factorisation $uv_n = x_n y_n z_n$ above can cross at most two words from $u_{n,1}, u_{n,2}, u_{n,3}$. It means that $x_n y'_n z_n \notin L$ for sufficiently large n, a contradiction.

The "if" part is achieved in a similar way as the regular case: we can construct a pushdown automaton $\mathcal{A}_{T,S}$, where S is the maximum pumpable set in T, so that $L_T = L(\mathcal{A}_{T,S})$. The only difference is that $\mathcal{A}_{T,S}$ uses its stack for checking the consistency the occurrences of two linearly independent occurrence vectors. $\mathcal{A}_{T,S}$ achieves it as some pushdown automaton recognises $L_A(a, b)$. □

6 Conclusion and Future Work

In this paper, we provided decidable, necessary and sufficient conditions of the regularity and context-freeness for WMIX languages by using the notion of dimensions. Complexity issues on these problems (tight lower/upper bounds, more efficient algorithm, *etc.*) are untouched and could be future work.

The author's main interest is how to generalise the main result into more richer language classes, *e.g.*, UnCA languages [11]. From WMIX languages (represented by de Bruijn graphs and diagonals $\{n \cdot \mathbf{1} \mid n \in \mathbb{N}\}$) into UnCA languages (represented by unambiguous automata and semilinear sets), although we should modify the notion of dimensions and some part of the proof strategy, the author conjectures that the context-freeness is still decidable for UnCA languages.

Acknowledgement: The author would like to thank Thomas Finn Lidbetter for telling me this topic in DLT 2018. Special thanks also go to my colleague

Fazekas Szilard whose helpful discussion were an enormous help to me. The author also thank to anonymous reviewers for many valuable comments. This work was supported by JSPS KAKENHI Grant Number JP19K14582.

References

1. Joshi, A., Vijay-Shanker, K., Weir, D. The Convergence of Mildly Context-sensitive Grammar Formalisms. Foundational Issues in Natural Language Processing (1991) 31–82
2. Marsh, W.: Some conjectures on indexed languages. Abstract appears in Journal of Symbolic Logic **51**(3) (1985) 849
3. Boullier, P.: Chinese numbers, mix, scrambling, and concatenation grammars range. In: EACL 1999, 9th Conference of the European Chapter of the Association for Computational Linguistics, June 8-12, 1999, University of Bergen, Bergen, Norway, The Association for Computer Linguistics (1999) 53–60
4. Kanazawa, M., Salvati, S.: MIX is not a tree-adjoining language. In: The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 1: Long Papers, The Association for Computer Linguistics (2012) 666–674
5. Salvati, S.: MIX is a 2-MCFL and the word problem in $\mathbb{Z}^2$ is captured by the IO and the OI hierarchies. Journal of Computer and System Sciences **81**(7) (2015) 1252–1277
6. Parikh, R.: On context-free languages. J. ACM **13**(4) (1966) 570–581
7. Colbourn, C.J., Dougherty, R.E., Lidbetter, T.F., Shallit, J.O.: Counting subwords and regular languages. In: Developments in Language Theory - 22nd International Conference, DLT 2018, Tokyo, Japan, September 10-14, 2018, Proceedings. Volume 11088 of Lecture Notes in Computer Science., Springer (2018) 231–242
8. Mateescu, A., Salomaa, A., Salomaa, K., Yu, S.: A sharpening of the Parikh mapping. Theoretical Informatics and Applications **35**(6) (2001) 551–564
9. Mateescu, A., Salomaa, A., Yu, S.: Subword histories and Parikh matrices. Journal of Computer and System Sciences **68**(1) (2004) 1–21
10. Seki, S.: Absoluteness of subword inequality is undecidable. Theoretical Computer Science **418** (2012) 116–120
11. Cadilhac, M., Finkel, A., McKenzie, P.: Unambiguous constrained automata. In Yen, H., Ibarra, O.H., eds.: Developments in Language Theory - 16th International Conference, DLT 2012, Taipei, Taiwan, August 14-17, 2012. Proceedings. Volume 7410 of Lecture Notes in Computer Science., Springer (2012) 239–250
12. Ginsburg, S.: The Mathematical Theory of Context-Free Languages. McGraw-Hill, Inc. (1966)
13. Kszonyi, L.: A pumping lemma for DLI-languages. Discrete Mathematics **258**(1) (2002) 105–122
14. Leroux, J., Penelle, V., Sutre, G.: The context-freeness problem is coNP-complete for flat counter systems. In: Automated Technology for Verification and Analysis - 12th International Symposium, ATVA 2014, Sydney, NSW, Australia, November 3-7, 2014, Proceedings. Volume 8837 of Lecture Notes in Computer Science., Springer (2014) 248–263
15. Schwer, S.R.: The context-freeness of the languages associated with vector addition systems is decidable. Theoretical Computer Science **98**(2) (1992) 199–247

16. de Luca, A., Varricchio, S.: Finiteness and Regularity in Semigroups and Formal Languages. Monographs in Theoretical Computer Science. An EATCS Series. Springer (1999)
17. Sin'ya, R.: Note on the infiniteness of $L(w_1, \ldots, w_k)$. CoRR **abs/1812.02600** (2018)
18. Takahashi, M.: A characterization of the derivation trees of a context-free grammar and an intercalation theorem. Master's thesis, University of Pennsylvania (1970)
19. Sin'ya, R.: Simple proof of Parikh's theorem à la takahashi. CoRR **abs/1909.09393** (2019)