Reducing the Ambiguity of Parikh Matrices

Jeffery Dick, Laura K. Hutchinson, Robert Mercaş, and Daniel Reidenbach

Department of Computer Science, Loughborough University, United Kingdom
{J.Dick, L.Hutchinson, R.G.Mercas, D.Reidenbach}@lboro.ac.uk

Abstract. The Parikh matrix mapping allows us to describe words using matrices. Although compact, this description comes with a level of ambiguity since a single matrix may describe multiple words. This work looks at how considering the Parikh matrices of various transformations of a given word can decrease that ambiguity. More specifically, for any word, we study the Parikh matrix of its Lyndon conjugate as well as that of its projection to a smaller alphabet. Our results demonstrate that ambiguity can often be reduced using these concepts, and we give conditions on when they succeed.

Keywords: Combinatorics, Parikh matrix, Ambiguity, Lyndon conjugate

1 Introduction

An approach for a more compact representation of data can be provided by histograms, which are also a well established statistical tool used in a wide range of applications. The concept of a Parikh vector [15] represents a type of such histograms that is specific to the analysis of sequences of symbols (or: words), considering the number of occurrences of each letter that exists in a word.

Parikh vectors can be easily computed and are guaranteed to be logarithmic in the size of the word they represent, but they are ambiguous; that is, multiple words typically share the same Parikh vector. Following this, in [14] the authors look at a refinement of the vector notion which is meant to reduce this ambiguity, and introduce an extension for it in the form of a Parikh matrix. A Parikh matrix not only contains the Parikh vector of the word, but also information regarding some of the word's (scattered) subwords. Such a matrix has the same asymptotic compactness as a Parikh vector and is associated to a significantly smaller number of words. However, it does not normally remove ambiguity entirely.

The bulk of the work done on the Parikh matrix mapping concerns the ambiguity that Parikh matrices exhibit. A lot of effort is spent on identifying an alternative to the Parikh matrix concept that would make a mapping from a word injective, or less ambiguous in general [1,2,8,9,10,11,18]. These include even more refined versions of the matrices by inclusion of polynomials, various extensions on the mappings, or both. For Parikh matrices explicitly, due to the difficulty arising from this ambiguity, the primary focus was on investigating this property on binary [4,5,6,7,17] and ternary [3,13,16,19] alphabets, leaving alphabets of size greater than three relatively unexplored.

In terms of reducing the ambiguity of a *word*, the investigation was based on either gathering more information about the specific word by altering the order of the alphabet, known as the dual order [6,14], or by considering the reverse image of the word [6]. Hence an under-studied aspect that may reduce the ambiguity of a matrix concerns the information acquired by altering the word itself, or considering other alterations of the alphabet. In this work we present and investigate two different methods that reduce the ambiguity of the original Parikh matrices in the form of $\mathbb{P}$-Parikh matrices and $\mathbb{L}$-Parikh matrices.

The first of the two transformations, the $\mathbb{P}$-Parikh matrix mapping, considers the Parikh matrices associated to a projection morphism of the initial word, where the considered alphabet is reduced to the subset of the alphabet used within the defined transformation. These represent a particular case of the extended mapping presented in [18], where we only consider a subset of the original alphabet. For example, consider the words *abcaabaac* and *abacabcaa*. It is easy to see that both share the same number of letters, and subwords *ab*, *bc* and *abc*, respectively, making their Parikh matrices equal and therefore ambiguous. The $\mathbb{P}$-Parikh matrices associated to them with respect to $\{a, c\}$ consider the number of subwords *ac*, which is 6 in the former, but only 5 in the latter of the words. Hence, there exist $\mathbb{P}$-Parikh matrices not shared by the words.

We show that, using $\mathbb{P}$-Parikh matrices, we can reduce the ambiguity of the vast majority of words. We also explore when $\mathbb{P}$-Parikh matrices do not reduce ambiguity, as well as provide some insight into the types of words that cannot be uniquely described by a $\mathbb{P}$-Parikh matrix.

However, since $\mathbb{P}$-Parikh matrices are defined for a subset of the initial alphabet, they prove useless when dealing with binary sequences. We therefore consider an alternative transformation of words: the Lyndon conjugate, first introduced in [20], which is defined as the lexicographically smallest circular rotation of a word. Lyndon conjugates were used previously as a tool for ambiguity reduction. In [17], the authors define the Lyndon image of a Parikh matrix as the lexicographically smallest word describing such a matrix. Hence every Parikh matrix has exactly one distinct Lyndon image, which therefore allows each Parikh matrix to be described uniquely. In the context of this paper, we use the Lyndon conjugate differently, i. e., we consider the Parikh matrix of the Lyndon conjugate of a word, and we call the resulting matrix the $\mathbb{L}$-Parikh matrix of the original word.

Consider the Parikh matrix of the Lyndon conjugates of the two previously given words. Observe that *aabaacabc* has 7 occurrences of *ab*, whereas *aaabacabc* has 8, making their Parikh matrices different. Hence, the ambiguity of their Parikh matrix can be reduced using $\mathbb{L}$-Parikh matrices.

While $\mathbb{L}$-Parikh matrices are a useful concept for any alphabet size, we focus on the cases where they reduce ambiguity in the binary alphabet and show that this happens in most cases. We give specific conditions of when $\mathbb{L}$-Parikh matrices do not help reduce the ambiguity of the given word, and investigate the words for which these criteria apply. This leads us to our main result of

the paper, a characterisation of words whose ambiguity can be reduced using $\mathbb{L}$-Parikh matrices.

We end this section with a brief breakdown of our paper. In Section 2 we present some basic definitions and notions. Section 3 examines the first of the two notions we introduce, the $\mathbb{P}$-Parikh matrix, establishing conditions for when they can or cannot achieve ambiguity reduction. In Section 4, we study equivalent questions for $\mathbb{L}$-Parikh matrices, largely focusing on binary alphabets in some cases. We end our paper with conclusions as well as directions for future work.

2 Preliminaries

It is assumed the reader is familiar with the basics of combinatorics on words. If needed, [12] can be consulted. Throughout this paper, $\mathbb{N}$ refers to the set of natural numbers starting with 1.

We refer to a string of arbitrary letters as a *word* which is formed by concatenation of letters. The set of all letters used to create our words is called an *alphabet*. We represent an *ordered alphabet* as $\Sigma_k = \{a_1 < \cdots < a_k\}$, where $k \in \mathbb{N}$ is the *size* of the alphabet, and by convention a_i is the ith letter in the Latin alphabet. Whenever the alphabet size is irrelevant or understood, we omit this from notation using only Σ. All alphabets referred to in this paper have an order imposed on them.

We define the concatenation of two words u and v as uv. The *length* of a word is the total number of not necessarily distinct letters it contains and the *empty word* of length zero is denoted ε. The *Kleene star*, denoted *, is the operation that, once applied to a given alphabet, generates the set of all finite words that result from concatenating any words in that alphabet. Further, we denote the i^{th} letter in a word w as $w[i]$.

The *reversal* of a word, denoted *rev*, is defined as $rev(w) = w[m]w[m-1]\cdots w[1]$, where $w = w[1]w[2]\cdots w[m]$ is a word with $w[i] \in \Sigma$. We say that a *factor* v is in w if and only if w can be written as $w = w_1 v w_2$, where $w_1, w_2 \in \Sigma^*$. We say that $u = u[1]u[2]\cdots u[m]$ is a *subword* of v if we have a factorisation $v = v_0 u[1] v_1 u[2] \cdots v_{m-1} u[m] v_m$ where $v_0, \ldots, v_m \in \Sigma^*, u[1], \ldots, u[m] \in \Sigma$. We use $|v|_u$ to denote the number of distinct occurrences of u as a subword in v.

The *Parikh vector* [15] ϕ associated with a word w is obtained through a mapping $\phi : \Sigma^* \to \mathbb{N}^k$, defined as $\phi(w) = [|w|_{a_1}, |w|_{a_2}, \ldots, |w|_{a_k}]$. For a matrix M of size $k \times k$, the *j-diagonal* is defined as all elements of M that are in the position $M_{i,i+j}$ for $i = 1, \ldots, k-j$. A word is *associated* with a matrix, called its Parikh matrix, if the matrix is obtained from that word following the process detailed in the following explanatory definition. For a technical version of the definition we refer to [14].

Definition 1 (Explanatory). *Let $w \in \Sigma_k^*$. The* Parikh matrix*, denoted $\Psi(w)$, that w is associated with has size $(k+1) \times (k+1)$. The diagonal of the matrix is populated with 1's and all elements below it are 0. The count of all subwords that consist of consecutive letters in Σ_k and are of length n in the word are found on the n-diagonal, for $1 \leq n \leq k$.*

One notion we introduce in this paper relies on a change in alphabet size. As such, to emphasise the size n of the alphabet used for a Parikh matrix, we write $\Psi_n(w)$. We say that a Parikh matrix *describes* a word if the word is associated to the matrix. Notice that due to the associativity of matrix multiplication, the Parikh matrix of a word can be constructed from the Parikh matrices of its factors. For a word $w = u_1u_2$, we have $\Psi_n(w) = \Psi_n(u_1)\Psi_n(u_2)$.

Example 1. Consider the word $w = abca$ defined over the alphabet $\Sigma_3 = \{a < b < c\}$. Then by definition our Parikh matrix is of size 4×4 and we have

$$\Psi(abca) = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

For the rest of this work we refine our notation for a Parikh matrix where we remove the elements not depending on the associated word. By definition a Parikh matrix is an upper triangular matrix with 1's on the diagonal regardless of the word described. For aesthetics, removing the redundant part leaves us with a triangular structure that holds the same information as the original matrix,

$$\Psi(abca) = \begin{pmatrix} 2 & 1 & 1 \\ & 1 & 1 \\ & & 1 \end{pmatrix}. \qquad \triangleleft$$

Two words w and w' are *conjugates* if we can write $w = uv$ and $w' = vu$. For a word w, we say that the *conjugacy class of* w, denoted $C(w)$ is the class of all of its possible conjugates. A *conjugacy class is associated to a Parikh matrix* if at least one word belonging to that class is associated to the matrix.

Example 2. The matrix $\begin{pmatrix} 4 & 4 \\ & 2 \end{pmatrix}$ has only the words $aabbaa, abaaba, baaaab$ associated to it. The words $aabbaa$ and $baaaab$ are members of the same conjugacy class, while $abaaba$ belongs to a different conjugacy class. Hence this matrix has two conjugacy classes associated to it. $\triangleleft$

A Parikh matrix can be associated to multiple words, as seen above, although cases exist where a matrix describes a single word, e. g., $aabb$ is the unique word associated to $\begin{pmatrix} 2 & 4 \\ & 2 \end{pmatrix}$. We say that two words are *amiable* if they are associated to the same Parikh matrix. If two or more words are associated to a single Parikh matrix, we say that the matrix is ambiguous. Later in this paper, we reduce the ambiguity of a word using both its Parikh matrix and the Parikh matrix of an altered form of that word to describe it. As such, we introduce a formal definition of the *ambiguity* that multiple functions may have based on the set of all words that satisfy all functions. We are then able to use this when considering the ambiguity of the notions we introduce later.

Definition 2. *For a word w and functions $f_1, \ldots, f_n$ we define $\mathcal{A}(w, f_1, \ldots, f_n) = \{v | f_i(v) = f_i(w)$ for $1 \leq i \leq n\}$. If $|\mathcal{A}(w, f_1, \ldots, f_n)| = 1$, then we call w* unambiguous *on $f_1, \ldots, f_n$, and say that $f_1(w), \ldots, f_n(w)$* uniquely define *w. However, if $|\mathcal{A}(w, f_1, \ldots, f_n)| > |\mathcal{A}(w, f_1, \ldots, f_m)|$ for $m > n$ and functions $f_{n+1}, \ldots, f_m$, then we say that $f_{n+1}, \ldots, f_m$* reduce the ambiguity of *w on $f_1, \ldots, f_n$.*

Observe that we always have $|\mathcal{A}(w, f_1, \ldots, f_n)| \geq |\mathcal{A}(w, f_1, \ldots, f_m)|$. Furthermore, if $|\mathcal{A}(w, f_1, \ldots, f_n)| = |\mathcal{A}(w, f_1, \ldots, f_m)| = 1$, then $\mathcal{A}(w, f_1, \ldots, f_n)$ is unambiguous and it is not possible to further reduce ambiguity.

First we introduce the $\mathbb{P}$-Parikh matrix. This matrix is in essence the Parikh matrix of a projection of a word, and represents a particular case of the extension of the Parikh matrix mapping presented in [19]. For $n \in \mathbb{N}$, $w \in \Sigma_n^*$ and $S \subset \Sigma_n$, the $\mathbb{P}$-Parikh matrix of w with respect to S is defined as follows.

Definition 3. *For $m, n \in \mathbb{N}$ with $1 \leq m \leq n$, let $S \subset \Sigma_n$ such that $S = \{a_{k_1}, a_{k_2}, \ldots a_{k_m}\}$, where $0 < k_1 < \cdots < k_m \leq n$. We define the $\mathbb{P}$-Parikh matrix of the word w with respect to S as $\Psi_n^S(w) := \Psi_{|S|}(\pi_S(w))$, where the morphism $\pi : \Sigma_n^* \to \Sigma_m^*$ is defined as*

$$\pi_S(a_j) := \begin{cases} a_i & : a_j = a_{k_i} \\ \varepsilon & : a_j \notin S \end{cases}.$$

To gain some intuition about the above definition, consider an example.

Example 3. Let $\Sigma_5 = \{a, b, c, d, e\}$, $S = \{a, d, e\}$, and $w = bacbebda$. For the index sequence of S, since a is the lexicographically smallest letter in S, we obtain $k_1 = 1$, $k_2 = 4$ and $k_3 = 5$. Hence $\pi_S(a) = a$, $\pi_S(d) = b$ and $\pi_S(e) = c$.

With the transformation defined we apply this to the word, and calculate the corresponding $\mathbb{P}$-Parikh matrix as the Parikh matrix of the transformed word,

$$\pi_S(w) = \pi_S(b)\pi_S(a)\pi_S(c)\pi_S(b)\pi_S(e)\pi_S(b)\pi_S(d)\pi_S(a) = \varepsilon a \varepsilon \varepsilon c \varepsilon b a = acba$$

$$\Psi_5^{\{a,d,e\}}(bacbebda) = \Psi_3(\pi_{\{a,d,e\}}(bacbebda)) = \Psi_3(acba) = \begin{pmatrix} 2 & 1 & 0 \\ & 1 & 0 \\ & & 1 \end{pmatrix}. \quad \triangleleft$$

The *Lyndon conjugate* of a word is the conjugate that is lexicographically smallest based on the order on the alphabet. The Lyndon conjugate of a word w is denoted $L(w)$. In an attempt to reduce the ambiguity of Parikh matrices, we modify the original Parikh matrix mapping to gain more information about a given word. Next, we introduce the $\mathbb{L}$-Parikh matrix associated to a word.

Definition 4. *Given a word w, we define its* $\mathbb{L}$-Parikh matrix, *Ψ_L, as the Parikh matrix associated with its Lyndon conjugate, $L(w)$.*

It was shown in [4] that there exist transformations that, when applied to a word, create a new word that is amiable with the original. For non-binary alphabets, a Type 1 transformation is given.

Lemma 1 ([4]). *Let $w, w' \in \Sigma_n^*$ with $n \geq 3$. Then w transforms into w' using a* Type 1 *transformation if $w = u_1 a_i a_j u_2$ and $w' = u_1 a_j a_i u_2$, where $u_1, u_2 \in \Sigma_n^*$, $a_i, a_j \in \Sigma_n$, and $|i - j| \geq 2$.*

For binary alphabets, a second type of transformation is described, referred to as a Type 2, that allows us to check if certain words are amiable without constructing their matrices.

Lemma 2 ([4]). *Let $w, w' \in \Sigma_2^*$. Then w transforms into w' through a* Type 2 *transformation if $w = x a_1 a_2 y a_2 a_1 z$ and $w' = x a_2 a_1 y a_1 a_2 z$, or vice-versa, where $x, y, z \in \Sigma_2^*$ and $a_1, a_2 \in \Sigma_2$.*

3 $\mathbb{P}$-Parikh Matrices

In this section, we examine when and how much $\mathbb{P}$-Parikh matrices reduce the ambiguity of a given word. When we refer to a reduction in ambiguity using $\mathbb{P}$-Parikh matrices, we mean that the number of words described by the original Parikh matrix and their respective $\mathbb{P}$-Parikh matrices is strictly less than the total number of words described by the original Parikh matrix alone, i. e., $\left|\mathcal{A}\{w, \Psi_n, \Psi_n^S\}\right| < |\mathcal{A}\{w, \Psi_n\}|$, for some $S \subset \Sigma_n$. First we present an example of $\mathbb{P}$-Parikh matrices removing the ambiguity of a Parikh matrix entirely.

Example 4. Consider the word $w = abca$ from Example 1, which is amiable with the word $w' = abac$ and no others. Then we choose our set $S = \{a, c\}$, and get that: $\Psi_3^{\{a,c\}}(w) = \Psi_2(aba) = \left\langle \begin{smallmatrix} 2 & 1 \\ & 1 \end{smallmatrix} \right\rangle$ and $\Psi_3^{\{a,c\}}(w') = \Psi_2(aab) = \left\langle \begin{smallmatrix} 2 & 2 \\ & 1 \end{smallmatrix} \right\rangle$. Thus w and w' have different $\mathbb{P}$-Parikh matrices and we can uniquely describe them. ◁

We first introduce some terms that are useful when describing how effective a given $\mathbb{P}$-Parikh matrix is at reducing ambiguity.

Definition 5. *Given a word $w \in \Sigma_n^*$, we call $\Psi(w)$* $\mathbb{P}$-distinguishable *if either $|\mathcal{A}(w, \Psi)| = 1$ or there exists a word $u \in \Sigma_n^*$ and a set $S \subset \Sigma_n$ such that $\Psi(w) = \Psi(u)$ and $\Psi_n^S(w) \neq \Psi_n^S(u)$. In the latter case we say that w and u are* $\mathbb{P}$-distinct. *Furthermore, we call w* $\mathbb{P}$-unique *if there exist sets $S_1, S_2, \ldots, S_m \subset \Sigma_n$ such that $|\mathcal{A}(w, \Psi, \Psi_n^{S_1}, \Psi_n^{S_2}, \ldots, \Psi_n^{S_m})| = 1$.*

Now we use these terms to examine words whose ambiguity can be reduced using $\mathbb{P}$-Parikh matrices, namely those that contain any length two factor where those two letters are not equal or consecutive in the alphabet.

Proposition 1. *For any word $w \in \Sigma_n^*$ with a factor $a_i a_j$ where $|i - j| > 1$, we have that $\Psi(w)$ is $\mathbb{P}$-distinguishable.*

Proof. Since $|i - j| > 1$, if $w = u_1 a_i a_j u_2$ where $u_1, u_2 \in \Sigma_n^*$, then $w' = u_1 a_j a_i u_2$ is also associated to w, following Lemma 1. Without loss of generality, take $S = \{a_i < a_j\}$. Then $\Psi_n^S(w) \neq \Psi_n^S(w')$, since $|w|_{a_i a_j}$ and $|w'|_{a_i a_j}$ are elements in $\Psi_n^S(w)$ and $\Psi_n^S(w')$, respectively, and $|w|_{a_i a_j} \neq |w'|_{a_i a_j}$. □

It is simple to identify words that have such factors by comparing adjacent positions in the word. We can use this to find a lower bound for the proportion of words that are uniquely identified for a given alphabet and word length.

Proposition 2. *The number of words of length m in Σ_n that are reduced in ambiguity by $\mathbb{P}$-Parikh matrices is bounded below by $(n^m) - (n \times 3^{m-1})$.*

Notice especially that as n and m get larger, the proportion of words which are reduced in ambiguity by $\mathbb{P}$-Parikh matrices also gets larger. We therefore conclude that the use of $\mathbb{P}$-Parikh matrices reduces ambiguity for a larger ratio of words for bigger alphabets rather than smaller.

There also exist words for which $\mathbb{P}$-Parikh matrices do not reduce ambiguity. Our following result says that if our choice of a subset consists of only consecutive letters of the initial alphabet, the $\mathbb{P}$-Parikh matrices are not $\mathbb{P}$-distinguishable.

Remark 6 *If all elements of the set $S \subset \Sigma_n$ are consecutive in the alphabet Σ_n, then $|\mathcal{A}(w, \Psi_n)| = |\mathcal{A}(w, \Psi_n, \Psi_n^S)|$.*

The result of Remark 6 strengthens the one of Proposition 1 by telling us that the ambiguity of words defined over binary alphabets is not reducible by $\mathbb{P}$-Parikh matrices.

Corollary 1. *There does not exist a Parikh matrix that describes binary words whose ambiguity can be reduced by $\mathbb{P}$-Parikh matrices.*

Furthermore, there exist non-binary words for which $\mathbb{P}$-Parikh matrices do not remove ambiguity, namely those that are not $\mathbb{P}$-unique. Finally, we end this section by giving two classes of words which are not uniquely described by $\mathbb{P}$-Parikh matrices, no matter how we choose the set S.

Proposition 3. *Take two words $w, w' \in \Sigma_n^*$ with the form $w = u_1 a_i a_j v a_j a_i u_2$ and $w' = u_1 a_j a_i v a_i a_j u_2$, where $a_i \leq a_j \in \Sigma_n$ and $u_1, u_2 \in \Sigma_n^*$. If $v \in \{a_k \in \Sigma_n | a_i \leq a_k \leq a_j\}^*$, then for all $S \subseteq \Sigma_n$, we have $\Psi_n^S(w) = \Psi_n^S(w')$.*

Proof. Firstly, if $a_i = a_k = a_j$, equivalence follows, as $w = w'$. Now, let $a_i < a_j$.

In the case where S contains either a_i or a_j, then $\pi_S(w) = \pi_S(w')$ since a_i and a_j are the only letters that swap places in w' compared to w. Since $\pi_S(w) = \pi_S(w')$, clearly $\Psi_n^S(w) = \Psi_n^S(w')$ follows.

If $S = \{a_i, a_j\}$, then, $\pi_S(w)$ is a binary word and can be transformed via a Type 2 transformation, from Lemma 2, into $\pi_S(w')$, so $\Psi_n^S(w) = \Psi_n^S(w')$.

Next consider that $\{a_i, a_j\} \subset S$, $|S| > 2$, and S has no elements between a_i and a_j. Then $\pi_S(w)$=$\pi_S(u_1)a_i a_j a_j a_i \pi_S(u_2)$ and $\pi_S(w')$=$\pi_S(u_1)a_j a_i a_i a_j \pi_S(u_2)$. Using an extension from [3] of the Type 2 transformations we can transform $\pi_S(w)$ into $\pi_S(w')$, and get that $\Psi_n^S(w) = \Psi_n^S(w')$.

Finally, consider the case where S contains a_i, a_j, and at least one letter that comes lexicographically between a_i and a_j. Then, $\pi_S(w)$ can be transformed into $\pi_S(w')$ via two Type 1 transformations on a_i and a_j, since a_i and a_j are not lexicographically adjacent in S (see Lemma 1). □

The ideas from Proposition 3 give rise to another class of words that are not $\mathbb{P}$-unique, by loosening the condition on v and extending the length of the word.

Proposition 4. *Take two words of the form $w = u_1 a_i a_j v_1 a_j a_i a_j a_i v_2 a_i a_j u_2$, and $w' = u_1 a_j a_i v_1 a_i a_j a_i a_j v_2 a_j a_i u_2$, where $a_i < a_j \in \Sigma_n$ and $u_1, u_2, v_1, v_2 \in \Sigma_n^*$. Let $v_1 = v_1[1]v_1[2]\cdots v_1[x]$ and $v_2 = v_2[1]v_2[2]\cdots v_2[y]$. Then, w and w' are not $\mathbb{P}$-distinct if and only if $|v_1|_{a_\ell} = |v_2|_{a_\ell}$ for all $a_\ell \notin \{a_k | a_i \leq a_k \leq a_j\}$, and at least one of the following conditions is true:*

1. *$v_1, v_2 \in \{a_k | a_k \leq a_j\}^*$, and for $\ell < p$, if $v_2[p], v_2[\ell] \in \{a_k | a_k < a_i\}$, then $v_2[p] \leq v_2[\ell]$, and if $v_1[p], v_1[\ell] \in \{a_k | a_k < a_i\}$, then $v_1[p] \geq v_1[\ell]$;*
2. *$v_1, v_2 \in \{a_k | a_k \geq a_i\}^*$, and for $\ell < p$, if $v_2[p], v_2[\ell] \in \{a_k | a_k > a_j\}$, then $v_2[p] \geq v_2[\ell]$, and if $v_1[p], v_1[\ell] \in \{a_k | a_k > a_j\}$, then $v_1[p] \leq v_1[\ell]$.*

In other words, the above statement says that two words are not $\mathbb{P}$-distinct if both v_1 and v_2 are defined on the subset of the alphabet which is either lexicographically bigger than a_i or smaller than a_j, and they share the same Parikh vector for the subset of letters which are not in between a_i and a_j. Furthermore, if $v_1 \in \{a_{i+1}, \ldots, a_n\}^*$, then all the letters which are lexicographically greater than a_j must occur in v_1 in decreasing lexicographical order and in v_2 in increasing order. On the other hand, if $v_1 \in \{a_1, \ldots, a_{j-1}\}^*$, then all the letters which are lexicographically smaller than a_i must occur in v_1 in increasing lexicographically order and in v_2 in decreasing lexicographical order.

4 $\mathbb{L}$-Parikh Matrices

Proposition 2 shows that in many cases, the set of words that share both a Parikh matrix and a $\mathbb{P}$-Parikh matrix is smaller than the set of those that share only a Parikh matrix. However, following Corollary 1 we also know that this never happens for binary alphabets. Hence we now study $\mathbb{L}$-Parikh matrices as an alternative method of ambiguity reduction. While they can be effective for any non-unary alphabet, we focus on binary alphabets specifically. We begin this section by explaining the motivation for choosing the Lyndon conjugate of a word and then build to our main result where we characterise words whose ambiguity is reduced by the use of $\mathbb{L}$-Parikh matrices.

As indicated by Definition 4, the concept of $\mathbb{L}$-Parikh matrices is based on a modification to a word that results in a change in the order of letters. The following theorem implies that the strategy of altering a word is not always a successful method of ambiguity reduction. Note that Ψ_{rev} refers to the Parikh matrix of the reversal of a word.

Theorem 1 ([4]). *For a word w, we have that $\mathcal{A}(w, \Psi) = \mathcal{A}(w, \Psi, \Psi_{\text{rev}})$.*

Unlike Theorem 1, $\mathbb{L}$-Parikh matrices use the conjugate of a word. The next proposition implies that such conjugates need to be chosen wisely.

Proposition 5. *Given words $v, w \in \Sigma^*$ with $\Psi(v) = \Psi(w)$, for any factorisations $v = v_1v_2$ and $w = w_1w_2$ such that $|v_2| = |w_2|$, we have that $\Psi(v_2v_1) = \Psi(w_2w_1)$ implies $\phi(v_2) = \phi(w_2)$. For Σ_2, the reverse direction also stands, namely $\phi(v_2) = \phi(w_2)$ implies $\Psi(v_2v_1) = \Psi(w_2w_1)$.*

Proof Outline. We can prove the statement that holds for every size alphabet by contradiction, by assuming that $\Psi(v) = \Psi(w), \Psi(v_2v_1) = \Psi(w_2w_1)$ and $\phi(v_2) \neq \phi(w_2)$. We examine the total number of ab subwords in v, w, v_2v_1 and w_2w_1 to obtain a set of equations. We then consider the total number of b's in v_2 and w_2 to find a contradiction within these equations.

For the statement that holds just for the binary alphabet we examine the total number of ab subwords in $v_2v_1, w_2w_1, v_1, v_2, w_1$ and w_2 and get a contradiction in the equations we obtain by initially assuming that $\phi(v_2) = \phi(w_2)$, $\Psi(v) = \Psi(w)$ and $\Psi(v_2v_1) \neq \Psi(w_2w_1)$. □

Below example shows that $|v_2| = |w_2|$ is necessary for Proposition 5.

Example 5. Consider the words $v = aabaabbb$ with $v_2 = aabbb$ and $w = aaabbabb$ with $w_2 = abb$. One can easily find that $\Psi(v_2v_1) = \Psi(w_2w_1) = \left\langle \begin{smallmatrix} 4 & 10 \\ & 4 \end{smallmatrix} \right\rangle$. Furthermore, we have that $\Psi(v) = \Psi(w)$, $\Psi(v_2v_1) = \Psi(w_2w_1)$ and $|v_2| \neq |w_2|$. However $\phi(v_2) \neq \phi(w_2)$, since $\phi(v_2) = [2, 3]$ and $\phi(w_2) = [1, 2]$, and therefore $|v_2| = |w_2|$ is a necessary condition in the context of Proposition 5. ◁

An example for the ternary alphabet where $\Psi(v_2v_1) \neq \Psi(w_2w_1)$ even though we have that $\Psi(v) = \Psi(w)$ and $\phi(v_2) = \phi(w_2)$ is given below. Note that if $\phi(v_2) = \phi(w_2)$, then we must also have $|w_2| = |v_2|$. Since any alphabet of size greater than 3 would rely on the result of the ternary alphabet always being true, we can deduce that the backwards direction from Proposition 5 only holds for the binary alphabet.

Example 6. Let $v = cbbaaabb$ and $w = cabbbaab$. We have that $\Psi(v) = \Psi(w)$. Now let $v_2 = aabb$ and $w_2 = baab$. Then we have that $|w_2| = |v_2|$ and $\phi(v_2) = \phi(w_2)$. Note that $\Psi(v_2) \neq \Psi(w_2)$, since $|v_2|_{ab} = 4$ and $|w_2|_{ab} = 2$. But this gives us $\Psi(v_2v_1) = \Psi(aabbcbba) \neq \Psi(baabcabb) = \Psi(w_2w_1)$. ◁

Proposition 5 shows that when looking for a modification that we can apply to a word to find a new and different Parikh matrix, we need to consider conjugates of amiable words where it is less likely that the Parikh vectors of their right factors are the same, i. e., conjugates obtained by shifting the original words a different number of times, respectively.

Let us now consider how using $\mathbb{L}$-Parikh matrices reduces ambiguity. The rest of this section ignores any word w where $|\mathcal{A}(w, \Psi)| = 1$, since there is no ambiguity to be reduced here. For a word w, we calculate $\Psi(w)$ and $\Psi_L(w)$ and use both of these matrices to describe the original word. The ambiguity of a word w, with respect to its Parikh and $\mathbb{L}$-Parikh matrices, according to Definition 2, is the total number of words that share a Parikh matrix and an $\mathbb{L}$-Parikh matrix with w, namely $|\mathcal{A}(w, \Psi, \Psi_L)|$. We use the next definitions and propositions to build to our main result where we characterise binary words whose ambiguity is reduced using $\mathbb{L}$-Parikh matrices. In line with Definition 5 we introduce the following definitions.

Definition 7. *Given a word $w \in \Sigma^*$, we call $\Psi(w)$* $\mathbb{L}$-distinguishable *if either $|\mathcal{A}(w, \Psi)| = 1$ or there exists a word $u \in \Sigma^*$ with $\Psi(w) = \Psi(u)$, such that $\Psi_L(w) \neq \Psi_L(u)$. In the latter case we say that w and u are* $\mathbb{L}$-distinct. *A word w is* $\mathbb{L}$-unique *if $|\mathcal{A}(w, \Psi, \Psi_L)| = 1$.*

Note that if w and v are $\mathbb{L}$-*distinct*, then $\mathcal{A}(w, \Psi) = \mathcal{A}(v, \Psi)$ and $\mathcal{A}(w, \Psi, \Psi_L) \neq \mathcal{A}(v, \Psi, \Psi_L)$. The example below demonstrates the effectiveness of $\mathbb{L}$-Parikh matrices for ambiguity reduction.

Example 7. Consider the words $w = babbbaa$, $u = bbababa$ and $v = bbbaaab$ with $\Psi(w) = \Psi(u) = \Psi(v)$. However, for the conjugates $L(w) = aababbb$, $L(u) = abababb$ and $L(v) = aaabbbb$ we have that $\Psi_L(w) = \left\langle \begin{smallmatrix} 3 & 11 \\ & 4 \end{smallmatrix} \right\rangle$, $\Psi_L(u) = \left\langle \begin{smallmatrix} 3 & 9 \\ & 4 \end{smallmatrix} \right\rangle$, and $\Psi_L(v) = \left\langle \begin{smallmatrix} 3 & 12 \\ & 4 \end{smallmatrix} \right\rangle$. Thus their $\mathbb{L}$-Parikh matrices are all different and we can uniquely describe each of the words by using $\mathbb{L}$-Parikh matrices. ◁

$\mathbb{L}$-distinguishability is necessary for ambiguity reduction in this case.

Corollary 2. *For $w \in \Sigma^*$, $|\mathcal{A}(w,\Psi)|>|\mathcal{A}(w,\Psi,\Psi_L)|$ iff $\Psi(w)$ is $\mathbb{L}$-distinguishable.*

The above characterisation of ambiguity reduction leads us to investigate sufficient conditions for a matrix to be ambiguous, and therefore for any pair of words it describes not to be $\mathbb{L}$-distinct. Our next results consider the situations when the Parikh matrix of a word is not $\mathbb{L}$-distinguishable. We show that words that meet the criteria outlined in each proposition within the binary alphabet are rare either later in the paper or directly following the next proposition.

Proposition 6. *For a word $w \in \Sigma^*$, if all words in $\mathcal{A}(w,\Psi)$ belong to the same conjugacy class, then $\Psi(w)$ is not $\mathbb{L}$-distinguishable.*

Example 8. Let $w = aababa$ and $w' = abaaab$. These two words are amiable with each other and nothing else. Furthermore, $L(w) = aaabab = L(w')$, and since both words share a Lyndon conjugate, both words also share an $\mathbb{L}$-Parikh matrix. Therefore $\Psi(w)$ is not $\mathbb{L}$-distinguishable. ◁

Now we move on to explore, for binary alphabets, the case where all words in $\mathcal{A}(w,\Psi)$ belong to the same conjugacy class in more detail. Recall that $C(w)$ refers to the conjugacy class of w.

Proposition 7. *Let $w \in \Sigma_2^*$. Then $L(u) = L(w)$, for all $u \in \mathcal{A}(w,\Psi)$, if and only if $L(w) \in \{aabb, ababbb, aababb, aabbab, aaabab\}$.*

Proof Outline. The forwards direction is proven by examining every element of the conjugacy class of w. We can first prove that if $L(u) = L(w)$, for all $u \in \mathcal{A}(w,\Psi)$, then words in the conjugacy class of w are only amiable with other conjugates of w. We then show that this is only true when $L(w)$ is in the set $\{aabb, ababbb, aababb, aabbab, aaabab\}$. For this we define a *block of a letter* to be a unary factor of a word which is not extendable to the right or left and argue that applying a Type 2 transformation to any Lyndon conjugate that is not in the above set either alters the size of the block of a's at the start of the word, or changes the total number of blocks of a's in the word altogether. This therefore gives us a word that is amiable to, but not a conjugate of, the original.

The backwards direction is proven by finding the Parikh matrices of all conjugates of words in the set $\{aabb, ababbb, aababb, aabbab, aaabab\}$. We then find that the only words described by these matrices are these conjugates. □

We now look at the case where all words associated to a Parikh matrix are the Lyndon representatives of their respective conjugacy classes, which again makes this matrix not $\mathbb{L}$-distinguishable.

Proposition 8. *For a word $w \in \Sigma^*$, if $|\mathcal{A}(w,\Psi)| \geq 2$ and $\mathcal{A}(w,\Psi) = \mathcal{A}(w,\Psi_L)$, then $\Psi(w)$ is not $\mathbb{L}$-distinguishable.*

Example 9. The words $w = aaaabaabbb$ and $w' = aaaaabbabb$ are only amiable with each other, $\Psi(w) = \Psi(w')$, and both are the Lyndon representatives of their respective conjugacy classes. Therefore, $\Psi(w) = \Psi(w') = \Psi_L(w) = \Psi_L(w')$ and $\Psi(w)$ is not $\mathbb{L}$-distinguishable. ◁

For binary alphabets, we examine in greater detail when all words in $\mathcal{A}(w, \Psi)$ are the Lyndon representatives of their conjugacy classes. The next result provides a necessary and sufficient condition, and therefore the complete characterisation, for this case to occur for the binary alphabet.

Proposition 9. *Let $w \in \Sigma_2^*$. Then the following statements are equivalent.*

- *For all $u \in \mathcal{A}(w, \Psi)$, we have that $u = L(u)$.*
- *$w = a^*vb^*$ and for $n = |v|_{ba}$ we have that $|v|_a = 2n$ and $|v|_b = n + 1$.*

Proof Outline. To show that these two statements are equivalent, we begin by showing that the second statement implies the former. We do this by first showing that if a word is of the form $w = a^*vb^*$ and, for $n = |v|_{ba}$, we have that $|v|_a = 2n$ and $|v|_b = n + 1$, then $w = L(w)$, and next move on to prove that only words of this form are described by $\Psi(w)$. We prove that $w = L(w)$ by observing that $v = L(v)$. Adding more a's to the start of v and more b's to the end means that the Lyndon conjugate is still the word itself, and hence obtain $w = L(w)$. We prove that words of the form described in the second point are only amiable with each other by calculating the total number of ab subwords in v and extrapolating this to w.

To prove that the first statement implies the second, we use the fact that our words share a Parikh matrix and that they must begin with the largest number of consecutive a's in the word and end with at least one b. We also rewrite $w = a^+w_i'b^+$ where w_i' begins with the first occurrence of a b and ends with the last occurrence of an a in w, and examine the form that this must take given the fixed number of ab subwords we must have in w. This gives us the total number of a's and b's in a word relative to the total number of ba subwords. □

Next example shows how the above result can be used to identify the form of the words that always share a Parikh matrix with other Lyndon conjugates.

Example 10. Following Proposition 9, Lyndon representatives of different conjugacy classes share a Parikh matrix only if they are of the form a^*vb^*, where for $n = |v|_{ba}$ we have that $|v|_a = 2n$ and $|v|_b = n + 1$. Let us find all words of this form where $n = 3$. We begin by finding all binary words that contain 3 subwords ba. These are $baaa, baba$ and $bbba$. Next add a's to the front and b's to the end of each word, respectively, so that we have a total of 6 a's and 4 b's per word: $aaabaaabbb, aaaababab, aaaaabbbab$. Finally, any number of a's and b's can be added to the front and end of each word, respectively: $a^*aaabaaabbbb^*, a^*aaaababab^*, a^*aaaaabbbabb^*$. Hence we know that any word of this form is the Lyndon representative of its conjugacy class and shares a Parikh matrix with the two other words stated above. For example, $\Psi(a^2aaabaaabbbb^3) = \Psi(a^2aaaababab^3) = \Psi(a^2aaaaabbbabb^3) = \left\langle \begin{smallmatrix} 8 & 53 \\ & 7 \end{smallmatrix} \right\rangle$. ◁

Thus far, we presented sufficient conditions for two amiable words not to be $\mathbb{L}$-distinct. Our main result shows that these conditions are in fact also the necessary ones. The following lemmas are used in the proof of the final result, but are included here as they are also interesting results on their own. First lemma

tells us that if the Parikh vectors of the proper right factors of two amiable words are different, then the size of these factors must also be unequal.

Lemma 3. *Consider the words $w = w_1w_2 = xabybaz$ and $v = v_1v_2 = xbayabz$ with $w, v \in \Sigma_2$, such that $w_1, w_2, v_1, v_2 \neq \varepsilon$ and $w_2w_1 = L(w) \neq L(v) = v_2v_1$. If $\phi(w_2) \neq \phi(v_2)$, then $|w_2| \neq |v_2|$.*

Furthermore, if two amiable binary words are not the Lyndon representatives of their conjugacy classes, then to either of them we can apply a Type 2 transformation to obtain an amiable word whose Lyndon conjugate begins in a different position from the original one.

Lemma 4. *Let $w = w_1w_2 \in \Sigma_2^*$ with $L(w) = w_2w_1 \neq w$. If $|\mathcal{A}(w, \Psi)| \geq 2$, then there exists $u = u_1u_2 \in \mathcal{A}(w, \Psi)$, where $L(u) = u_2u_1$, such that $|u_2| \neq |w_2|$.*

Proof Outline. The statement can be proven by contradiction, by first assuming that the Lyndon conjugate of every word associated to $\Psi(w)$ begins in the same position within those words. We then show that for the Lyndon conjugate to begin at any position within a given binary word, it is possible to apply a Type 2 transformation to obtain a new word whose Lyndon conjugate begins in a different position. □

Next we show that all words that are conjugates of any word w such that $\mathcal{A}(w, \Psi) = \mathcal{A}(w, \Psi_L)$ are also amiable with a word that is not a conjugate of any of the words in $\mathcal{A}(w, \Psi)$.

Lemma 5. *Let $w, u, v \in \Sigma_2^*$, where $\mathcal{A}(w, \Psi) = \mathcal{A}(w, \Psi_L)$. For any $u \in C(w)$ there exists $v \in \mathcal{A}(u, \Psi)$ such that $\mathcal{A}(w, \Psi_L) \cap C(v) = \emptyset$.*

Proof Outline. This statement can be proven by considering every form that a word w can take, such that $\mathcal{A}(w, \Psi) = \mathcal{A}(w, \Psi_L)$, from Proposition 9 and then examining all conjugates of these words. We show that a Type 2 transformation can be applied to every conjugate to obtain a word that is not a conjugate of any word in our original set $\mathcal{A}(w, \Psi)$. □

We end this section by giving our main result that characterises all binary words whose Parikh matrix is not $\mathbb{L}$-distinguishable.

Theorem 2. *For Σ_2, a Parikh matrix is not $\mathbb{L}$-distinguishable if and only if any of the words it describes meet at least one of the following criteria:*

- $w \in \{aabb, ababbb, aababb, aabbab, aaabab, bbabbaaa, bbbaabaa\}$
- $w = a^*vb^*$ *and for* $n = |v|_{ba}$ *we have that* $|v|_a = 2n$ *and* $|v|_b = n + 1$

Proof Outline. For the set of words $B = \{bbabbaaa, bbbaabaa\}$, the forward direction is easily proven by finding these words' Parikh and $\mathbb{L}$-Parikh matrices, respectively. The backward direction is proved using the fact that for words $w, w' \in \Sigma_2^*$ such that w is the reverse of w' and $\mathcal{A}(w', \Psi) = \mathcal{A}(w', \Psi_L)$, then $w \in B$ if and only if $\mathcal{A}(w, \Psi) = \mathcal{A}(w, \Psi, \Psi_L)$.

For the rest of the words, the 'if' direction was mostly proven earlier when Propositions 6, 7, 8 and 9, describing these situations, were introduced.

The 'only if' direction is proven by first examining the consequences of Proposition 5, which tells us that two words are $\mathbb{L}$-distinct if their Lyndon conjugates begin in different positions, respectively. We use Lemmas 3 and 4 to conclude that no set of amiable binary words exists where the Lyndon conjugates of all words in the set begin in the same position of each word, respectively. Hence all Parikh matrices would be $\mathbb{L}$-distinguishable if it were not for some cases that arise as a result of us using the Lyndon conjugate. These cases are namely the ones where the set of amiable words are all Lyndon conjugates, are all members of the same conjugacy class, or are all conjugates of words whose Lyndon conjugates share a Parikh matrix. We showed in Propositions 7 and 9 that the first two cases are characterised by words of the form $w = a^*vb^*$ where for $n = |v|_{ba}$ we have that $|v|_a = 2n$ and $|v|_b = n + 1$, and by words where their Lyndon conjugate is in the set $\{aabb, ababbb, aababb, aabbab, aaabab\}$, respectively. We use Lemma 5 to conclude that no words exist such that the third case is true. □

5 Conclusion and Future Work

In this paper, we have shown that using $\mathbb{P}$-Parikh matrices and $\mathbb{L}$-Parikh matrices reduces the ambiguity of a word in most cases. From Corollary 1, we learn that $\mathbb{P}$-Parikh matrices cannot reduce the ambiguity of a Parikh matrix that describes words in a binary alphabet, but are very powerful when it comes to reducing the ambiguity of words in larger alphabets (Proposition 2). On the other hand, we find that $\mathbb{L}$-Parikh matrices reduce the ambiguity of most binary words, with the few exceptions from Theorem 2, which have all been shown to be rare occurrences within the binary alphabet. Thus, using both tools together leads to a reduction in ambiguity in most cases.

Going forward, we wish to characterise words that are described uniquely by both types of matrices, respectively, as well as quantifying the ambiguity reduction permitted by both notions. Theorem 2 tells us that there are very few binary words whose Parikh matrix ambiguity cannot be reduced by $\mathbb{L}$-Parikh matrices. Future research on $\mathbb{L}$-Parikh matrices could also include an analysis similar to the one done in Proposition 2.

Finally we present a conjecture on the types of words that might be described by a Parikh matrix that is $\mathbb{P}$-distinguishable. We know that the presence of a certain type of factor, described in Proposition 1, in a word means that its Parikh matrix is $\mathbb{P}$-distinguishable. This conjecture implies that the presence of this factor is the *only* way that the ambiguity of a word could be reduced by $\mathbb{P}$-Parikh matrices.

Conjecture 8 *For any word $w \in \Sigma_n^*$, if $\Psi(w)$ is $\mathbb{P}$-distinguishable, then there exists a word amiable with w which contains a factor $a_i a_j$, where $|i - j| > 1$.*

References

1. Alazemi, H.M.K., Černý, A.: Counting subwords using a trie automaton. Int J Found Comput S **22**(6), 1457–1469 (2011)
2. Alazemi, H.M.K., Černý, A.: Several extensions of the Parikh matrix L-morphism. J Comput Syst Sci **79**(5), 658–668 (2013)
3. Atanasiu, A.: Parikh matrix mapping and amiability over a ternary alphabet. Discrete Mathematics and Computer Science pp. 1–12 (2014)
4. Atanasiu, A., Atanasiu, R., Petre, I.: Parikh matrices and amiable words. Theoret Comput Sci **390**(1), 102–109 (2008)
5. Atanasiu, A., Martín-Vide, C., Mateescu, A.: Codifiable languages and the Parikh matrix mapping. J UCS **7**, 783–793 (2001)
6. Atanasiu, A., Martín-Vide, C., Mateescu, A.: On the injectivity of the Parikh matrix mapping. Fund Inform **49**(4), 289–299 (2002)
7. Atanasiu, A., Teh, W.C.: A new operator over Parikh languages. Int J Found Comput S **27**(06), 757–769 (2016)
8. Bera, S., Mahalingam, K.: Some algebraic aspects of Parikh q-matrices. Int J Found Comput S **27**(4), 479–500 (2016)
9. Egecioglu, Ö.: A q-matrix encoding extending the Parikh matrix mapping. Technical Report 14, Department of Computer Science at UC Santa Barbara (2004)
10. Egecioglu, Ö., Ibarra, O.H.: A matrix q-analogue of the Parikh map. In: 3rd Int Conf TCS. IFIP, vol. 155, pp. 125–138 (2004)
11. Egecioglu, Ö., Ibarra, O.H.: A q-analogue of the Parikh matrix mapping. In: Formal Models, Languages and Applications [this volume commemorates the 75th birthday of Prof. Rani Siromoney]. Ser Mach Percept Artif Intell, vol. 66, pp. 97–111 (2007)
12. Lothaire, M.: Combinatorics on words. Cambridge University Press (1997)
13. Mahalingam, K., Subramanian, K.G.: Product of Parikh matrices and commutativity. Int J Found Comput S **23**(01), 207–223 (2012)
14. Mateescu, A., Salomaa, A., Salomaa, K., Yu, S.: On an extension of the Parikh mapping. Turku Cent Comput Sci (2000)
15. Parikh, R.J.: On context-free languages. J ACM **13**(4), 570–581 (1966)
16. Poovanandran, G., Teh, W.C.: Strong (2·t) and strong (3·t) transformations for strong M-equivalence. Int J Found Comput S **30**(05), 719–733 (2019)
17. Salomaa, A., Yu, S.: Subword occurrences, Parikh matrices and Lyndon images. Int J Found Comput S **21** (2010)
18. Şerbănuţă, T.F.: Extending Parikh matrices. Theoret Comput Sci **310**(1-3), 233–246 (2004)
19. Şerbănuţă, V.N.: On Parikh matrices, ambiguity, and prints. Int J Found Comput S **20**(01), 151–165 (2009)
20. Širšov, A.I.: Subalgebras of free Lie algebras. Mat. Sbornik N.S. **33(75)**, 441–452 (1953)