M-equivalence of Parikh Matrix over a Ternary Alphabet

Joonghyuk Hahn, Hyunjoon Cheon, and Yo-Sub Han

Department of Computer Science, Yonsei University, Seoul 03722, Republic of Korea
{greghahn,hyunjooncheon,emmous}@yonsei.ac.kr

Abstract. The Parikh matrix, an extension of the Parikh vector for words, is a fundamental concept in combinatorics on words. We investigate M-unambiguity that identifies words with unique Parikh matrices. While the problem of identifying M-unambiguous words for a binary alphabet is solved using a palindromicly amicable relation, it is open for larger alphabets. We propose substitution rules that establish M-equivalence and solve the problem of M-unambiguity for a ternary alphabet. Our rules build on the principles of the palindromicly amicable relation and enable tracking of the differences of length-3 ordered scattered-factors. We characterize the set of M-unambiguous words and obtain a regular expression for the set.

Keywords: Parikh Matrix · Parikh Vector · M-unambiguity · M-equivalence

1 Introduction

Parikh [10] introduced a concept of mapping words to vectors. The resulting vector is called a Parikh vector, by counting the occurrences of each letter [6, 7]. Mateescu et al. [9] extended the Parikh vector to the Parikh matrix that captures more complex numeric properties, by considering occurrences of scattered-factors.

Given an ordered alphabet $\Sigma = \{a_1, a_2, \ldots, a_k\}$, a Parikh matrix M over Σ is a $(k+1) \times (k+1)$ upper triangular matrix, where its main diagonal fills with all 1's, the second diagonal counts the occurrences of length-1 scattered-factors, the third diagonal counts length-2 ordered scattered-factors, and so on. For instance, given a word $w = 00121$ over $\Sigma = \{0, 1, 2\}$, the Parikh vector of w is $(|w|_0, |w|_1, |w|_2) = (2, 2, 1)$ and the Parikh matrix of w is

$$\begin{pmatrix} 1 & |w|_0 & |w|_{01} & |w|_{012} \\ 0 & 1 & |w|_1 & |w|_{12} \\ 0 & 0 & 1 & |w|_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 4 & 2 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Note that the second diagonal (in red) is the Parikh vector of w. Parikh matrices provide a simple and intuitive approach that computes the occurrences of scattered-factors.

We say that two words u and v are M-equivalent if u and v have the same Parikh matrix [5, 9]. Atanasiu et al. [3] identified a family of binary words having the same Parikh matrix and characterized M-equivalence over binary words by the concept of palindromicly amicable property. This characterization on M-equivalence allows the identification of words with unique Parikh matrices—the injectivity problem [2, 12]. Specifically, given a word u, if there is a distinct M-equivalent word v, then u is M-ambiguous; otherwise, u is M-unambiguous [8]. The injectivity problem is to find M-unambiguous words over a given ordered alphabet. Mateescu and Salomaa [8] constructed a regular expression for M-unambiguous words and solved the injectivity problem over a binary alphabet. However, it has been a challenging problem to extend this result to a larger alphabet.

Researchers partially characterized M-equivalence and M-unambiguity over a ternary alphabet [1, 4, 13]. Şerbănuţă and Şerbănuţă [4] enumerated all M-unambiguous words and proposed patterns that identify M-unambiguous words over a ternary alphabet. However, the pattern regular expression is incorrect; it misses some M-unambiguous words such as *bcbabcbabc*. Nevertheless, their work has laid a foundation for further research on M-unambiguity and M-equivalence [1, 13]. However, a complete and simple characterization towards M-equivalence and M-unambiguity over a general alphabet remains elusive [11, 14]. Even for a ternary alphabet, a comprehensive characterization of M-equivalence such as palindromicly amicable property has been open for decades.

We propose substitution rules that maintain the occurrences of length-1 or -2 ordered scattered-factors and keep track of the occurrences of length-3 ordered scattered-factors. Our three substitution rules can represent all words over a given ternary alphabet. We introduce $\cong$-relation that establishes M-equivalence based on the substitutions and compute the language of M-unambiguous words over a ternary alphabet.

We explain some terms and notations in Section 2. We introduce substitution rules and an equivalence relation that characterizes M-equivalent words in Section 3. Based on the proposed rules and relations, we compute a regular expression for M-ambiguous words and characterize M-unambiguity in Section 4. We conclude our paper with a brief summary and a few questions in Section 5.

2 Preliminaries

Let $\mathbb{N}$ denote the set of all nonnegative integers and $\mathbb{Z}$ denote the set of all integers. We use $\binom{u}{k}$ to denote the *binomial coefficient* for $u \geq k \in \mathbb{N}$. An *alphabet* Σ_k is a finite set of k letters and $|\Sigma_k| = k$ is the number of letters in Σ_k. We use Σ generally when the alphabet size k is not important. Without loss of generality, we use nonnegative integers as alphabet letters (e.g., $\Sigma_3 = \{0, 1, 2\}$). A *word* u over Σ is a finite sequence of letters in Σ. Let $|u|$ be the length of u. The symbol λ denotes the *empty word* whose length is 0. Given a word w, we use w^R to denote its reversal; $w = a_1a_2 \cdots a_n$ and $w^R = a_na_{n-1} \cdots a_1$. The *Kleene star* Σ^* of an alphabet Σ is the set of all words over Σ. An ordered alphabet $(\Sigma_k, <)$ consists of an alphabet $\Sigma_k = \{a_1, a_2, \ldots, a_k\}$ and a strict total order $<$ on Σ_k. We often denote the ordered alphabet by $\Sigma_k = \{a_1 < a_2 < \cdots < a_k\}$. If a total order $<$ is clear in the context, we simply use Σ_k to denote an ordered alphabet.

Given two words u and $v \in \Sigma^*$, we say that v is a *factor* of u if $u = \alpha v \beta$ for some $\alpha, \beta \in \Sigma^*$. Similarly, we say that v is a *scattered-factor* of u if there exist $u_0, u_1, \ldots, u_n$ and $v_1, v_2, \ldots, v_n \in \Sigma^*$ such that $v = v_1 v_2 \cdots v_n$ and $u = u_0 v_1 u_1 \cdots u_{n-1} v_n u_n$. We denote by $|u|_v$ the number of distinct occurrences of a nonempty word v as a scattered-factor in u. For instance, if $u = 0110$ and $v = 01$, then v is both a factor and a scattered-factor of u and $|u|_v = 2$.

We now present definitions that are directly related to our problem on the Parikh matrix.

Definition 1. *Let $\Sigma_k = \{a_1 < a_2 < \cdots < a_k\}$ be an ordered alphabet. The Parikh mapping is a monoid morphism $\Psi : \Sigma_k^* \to \mathbb{N}^k$ defined as $\Psi(w) = (|w|_{a_1}, |w|_{a_2}, \ldots, |w|_{a_k})$. Then, $\Psi(w)$ for $w \in \Sigma_k^*$ is the Parikh vector of w.*

The extension of the Parikh mapping to the Parikh matrix mapping considers a (upper) *unitriangular matrices* of nonnegative integers. A unitriangular matrix is a square matrix $m = (m_{i,j})_{1 \le i,j \le k}$ such that (1) $m_{i,j} \in \mathbb{N}$, (2) $m_{i,j} = 0$ for all $1 \le j < i \le k$, and (3) $m_{i,i} = 1$ for all $1 \le i \le k$. The set of all unitriangular matrices of dimension $k \ge 1$ is denoted by $\mathcal{M}_k$.

Definition 2. *Let $\Sigma_k = \{a_1 < a_2 < \cdots < a_k\}$ be an ordered alphabet. The Parikh matrix mapping is a monoid morphism $\Psi_{\Sigma_k} : \Sigma_k^* \to \mathcal{M}_{k+1}$ defined as follows. For $a_t \in \Sigma_k$, if $\Psi_{\Sigma_k}(a_t) = (m_{i,j})_{1 \le i,j \le k+1}$, then $m_{i,i} = 1$ for $1 \le i \le k+1$, $m_{t,t+1} = 1$, and all the other entries are zero. Then, $\Psi_{\Sigma_k}(w)$ for $w \in \Sigma_k^*$ is the Parikh matrix of w.*

Proposition 1. [9, Theorem 3.1] *Let $\Sigma_k = \{a_1 < a_2 < \cdots < a_k\}$ be an ordered alphabet. We denote by $a_{i,j}$ the word $a_i a_{i+1} \cdots a_j$ for $1 \le i \le j \le k$. For $w \in \Sigma_k^*$, its Parikh matrix $\Psi_{\Sigma_k}(w)$ has the following properties:*

1. *$m_{i,j} = 0$, for all $1 \le j < i \le k+1$,*
2. *$m_{i,i} = 1$, for all $1 \le i \le k+1$,*
3. *$m_{i,j+1} = |w|_u$ where $u = a_{i,j}$ for all $1 \le i \le j \le k$.*

Note that the Parikh matrix $\Psi_\Sigma(w)$ for $w \in \Sigma^*$ satisfies the associativity of matrix multiplication and $\Psi_\Sigma(w)$ can be constructed from the Parikh matrices of factors of w. For instance when $w = uv$, we have $\Psi_\Sigma(w) = \Psi_\Sigma(u) \cdot \Psi_\Sigma(v)$.

Example 1. Consider $w = 0110$ over a binary alphabet $\Sigma_2 = \{0 < 1\}$. As an example for Proposition 1,

$$\Psi_{\Sigma_2}(0110) = \Psi_{\Sigma_2}(0)\Psi_{\Sigma_2}(1)\Psi_{\Sigma_2}(1)\Psi_{\Sigma_2}(0) = \begin{pmatrix} 1 & 2 & 2 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & |0110|_0 & |0110|_{01} \\ 0 & 1 & |0110|_1 \\ 0 & 0 & 1 \end{pmatrix}.$$

3 *M*-equivalence

We discuss words with the equivalent Parikh matrices. For instance, the following words u and $v \in \Sigma^*$ have the same Parikh matrix.

$$\Psi_{\Sigma_3}(u) = \begin{pmatrix} 1 & |u|_0 & |u|_{01} & |u|_{012} \\ 0 & 1 & |u|_1 & |u|_{12} \\ 0 & 0 & 1 & |u|_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & |v|_0 & |v|_{01} & |v|_{012} \\ 0 & 1 & |v|_1 & |v|_{12} \\ 0 & 0 & 1 & |v|_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \Psi_{\Sigma_3}(v).$$

This equivalence relation is called *M-equivalence* [5, 9].

Definition 3. *Given two words w and w' over an ordered alphabet Σ, we define w and w' to be M-equivalent if $\Psi_\Sigma(w) = \Psi_\Sigma(w')$, and denote it by $w \equiv_M w'$.*

Researchers have studied how the changes in a word affect its Parikh matrix and when the Parikh matrix does not change. Proposition 2 illustrates substitutions of factors that do not change the Parikh matrix over arbitrary alphabets.

Proposition 2. [3, Proposition 3.1] *Let $\Sigma_k = \{a_1 < a_2 < \cdots < a_k\}$ be an ordered alphabet and $1 \leq i, j \leq k$. Then, the following equations hold:*

1. *If $|i - j| \geq 2$, then $\Psi_{\Sigma_k}(a_i a_j) = \Psi_{\Sigma_k}(a_j a_i)$.*
2. *If $|i - j| = 1$, then $\Psi_{\Sigma_k}(a_i a_j a_j a_i) = \Psi_{\Sigma_k}(a_j a_i a_i a_j)$.*

Proposition 2 is a necessary condition to establish M-equivalence but is not sufficient because they are not applicable to every word such as 10101, which is M-equivalent to 01110. Atanasiu et al. [3] proposed *palindromicly amicable* property that identifies M-equivalent words over a binary alphabet.

Definition 4. [3] *Let $\Sigma_2 = \{0 < 1\}$. Two words $\alpha, \beta \in \Sigma_2^*$ are palindromicly amicable if the following two statements hold:*

1. *α and β are palindromes,*
2. *$\Psi(\alpha) = \Psi(\beta)$.*

For $x, y \in \Sigma_2^*$ over $\Sigma_2 = \{0 < 1\}$, $x \equiv_{pa} y$ if a nonempty factor $\alpha \in \Sigma_2^*$ of x and a nonempty factor $\beta \in \Sigma_2^*$ of y are palindromicly amicable. We denote by $\equiv pa^*$, the reflexive and transitive closure of $\equiv_{pa}$.

Proposition 3. [3, Proposition 3.4] *For $x, y \in \Sigma_2^*$ over $\Sigma_2 = \{0 < 1\}$,*

1. $\equiv_{pa}^*$ is an equivalence relation.
2. If $x \equiv_{pa}^* y$, then for all $u \in \Sigma_2^*$, $ux \equiv_{pa}^* uy$ and $xu \equiv_{pa}^* yu$.

Theorem 1. [3, Theorem 3.1] *For $\Sigma_2 = \{0 < 1\}$ and $x, y \in \Sigma_2^*$, $x \equiv_M y$ if and only if $x \equiv_{pa}^* y$.*

Theorem 1 is based on the palindromicly amicable relation between x and y. If we can compute y by substituting factors from x based on Proposition 2, then x and y are M-equivalent. This is because the relation keeps the same value of $|x|_{01}$ and the Parikh vector also does not change. Theorem 1, however, does not hold for an alphabet with three or more letters. For instance, let $\Sigma_3 = \{0 < 1 < 2\}$ and $x = 10122101$, $y = 01122110$. It is easy to see that $x \equiv_{pa}^* y$ but $x \not\equiv_M y$.[1]

We first consider the following conditions that are satisfied for words x and y over a ternary alphabet to be M-equivalent:

1. $\Psi(x) = \Psi(y)$,
2. $|x|_{01} = |y|_{01}$, $|x|_{12} = |y|_{12}$, and $|x|_{012} = |y|_{012}$.

Certain substitutions preserve the value of 01-,12-, and 012-occurrences, implying that the substitutions also do not change the Parikh matrix. We investigate what these substitutions are.

Proposition 4. [2, Theorem 13] *For $\Sigma_3 = \{0 < 1 < 2\}$, the following statements hold for $\alpha, \beta, u \in \Sigma_3^*$.*

1. *If $w = \alpha 02 \beta$ and $w' = \alpha 20 \beta$, then $w \equiv_M w'$.*
2. *If $w = \alpha 01 u 10 \beta$ and $w' = \alpha 10 u 01 \beta$ where $|u|_2 = 0$, then $w \equiv_M w'$.*
3. *If $w = \alpha 12 u 21 \beta$ and $w' = \alpha 21 u 12 \beta$ where $|u|_0 = 0$, then $w \equiv_M w'$.*

While Proposition 4 suggests useful substitution rules that preserve the Parikh matrix, the substitution rules are not applicable to all the words. We cannot apply the second rule to $w = \alpha 01 u 10 \beta$ such that $|u|_2 > 0$. Likewise, the third rule is not applicable to $w = \alpha 12 u 21 \beta$ such that $|u|_0 > 0$. For instance, we cannot apply any substitutions in Proposition 4 to an M-ambiguous word $u = 0101210121$. On the other hand, for w that we can apply substitutions in Proposition 4, we cannot enumerate all w' that are M-equivalent to w. For $u = 1002101112$, we cannot compute $u' = 0101210121$, which is M-equivalent to u by Proposition 4. If we design equivalence relations that maintain the same Parikh matrix for a given word $u \in \Sigma_3^*$, then any relations should preserver the value of $|u|_{01}$, $|u|_{12}$, and $|u|_{012}$. This leads us to design an equivalence relation that considers the following:

1. For all M-ambiguous words, the relation should be applicable.
2. Given an M-ambiguous word u, all M-equivalent words to u should be computed.

We relax the constraint that a single substitution rule should preserve the Parikh matrix value and allow the value of 012-occurrences to change. We suggest Proposition 4.

Proposition 5. *For $\Sigma_3 = \{0 < 1 < 2\}$, and $u, \alpha, \beta \in \Sigma_3^*$, the followings are substitution rules that satisfy $\Psi(w) = \Psi(w')$, $|w|_{01} = |w|_{01}$, and $|w|_{12} = |w'|_{12}$:*

1. *If $w = \alpha 02 \beta$ and $w' = \alpha 20 \beta$, then $|w|_{012} = |w'|_{012}$.*

[1] $x \equiv_{pa} 10211201 \equiv_{pa} 11200211 \equiv_{pa} 02111120 \equiv_{pa} y$

2. *If* $w = \alpha 01u10\beta$ *and* $w' = \alpha 10u01\beta$, *then* $|w|_{012} = |w'|_{012} + |u|_2$.
3. *If* $w = \alpha 12u21\beta$ *and* $w' = \alpha 21u12\beta$, *then* $|w|_{012} = |w'|_{012} - |u|_0$.

Proof. Three substitution rules satisfy $\Psi(w) = \Psi(w')$ because each rule does not change the numbers of 0's, 1's, and 2's. The first substitution rule does not change the numbers of 01's and 12's. The second and the third substitution rules change 01 to 10 (respectively, 12 to 21) and also change 10 to 01 (respectively, 21 to 12), which keeps the same numbers of 01's and 12's at the end. For the occurrences of 012, Proposition 5 can be deduced by computing $|w|_{012}$ and $|w'|_{012}$.

In the first substitution rule,

$$|w|_{012} = |\alpha 02\beta|_{012} = |\alpha|_{012} + |\beta|_{012} + |\alpha|_{01} \times (|02|_2 + |\beta|_2) + (|\alpha|_0 + |02|_0) \times |\beta|_{12},$$

$$|w'|_{012} = |\alpha 20\beta|_{012} = |\alpha|_{012} + |\beta|_{012} + |\alpha|_{01} \times (|20|_2 + |\beta|_2) + (|\alpha|_0 + |20|_0) \times |\beta|_{12}.$$

It is easy to verify that $|02|_0$ and $|02|_2$ are the same to $|20|_0$ and $|20|_2$, respectively. Therefore, we know that $|w|_{012} = |w'|_{012}$.

For the second substitution rule, the substitution only occurs in the factor $01u10$ in w. We only have to keep track of 012 occurrences in $01u10$ of w and $10u01$ of w'. While $|01u10|_{012} = |u|_{012} + |01|_0 \times |u|_{12} + |01|_{01} \times |u|_2$, after the substitution, $|10u01|_{012} = |u|_{012} + |10|_0 \times |u|_{12} + |10|_{01} \times |u|_2 = |u|_{012} + |u|_{12}$. Therefore, the second substitution rule reduces the occurrences of 012 by $|u|_2$. Similarly, we can show that the third substitution rule increases the occurrences of 012 by $|u|_0$. □

We employ the second and third substitution rules of Proposition 5 to keep track of the occurrences of 012 and furthermore, analyze when $|w|_{012} = |w'|_{012}$. For instance, given $w, w' \in \Sigma_3^*$, Figure 1 demonstrates that $|w|'_{012} = |w|_{012} - |\alpha|_2 + |\beta|_0$ when applied with the substitution rules of Proposition 5. Thus, when $|\alpha|_2 = |\beta|_0$, the Parikh matrices of w and w' are the same.

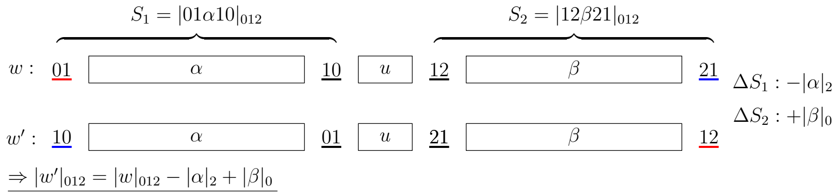
$S_1 = |01\alpha 10|_{012}$
$S_2 = |12\beta 21|_{012}$
w :
01
α
10
u
12
β
21
$\Delta S_1 : -|\alpha|_2$
$\Delta S_2 : +|\beta|_0$
w' :
10
α
01
u
21
β
12
$\Rightarrow |w'|_{012} = |w|_{012} - |\alpha|_2 + |\beta|_0$

Fig. 1. An illustration of substitutions maintaining M-equivalence for a word $w = 01\alpha 10u12\beta 21$.

Figure 1 demonstrates one of the four scenarios where a single substitution step involving two replacements maintains the identical $|w|_{012}$ values, thereby preserving the Parikh matrix. Additionally, there are cases where the swapping pairs overlap. Figure 2 further illustrates cases of alternating sequences, with 01 followed by 12 and 10 followed by 21.

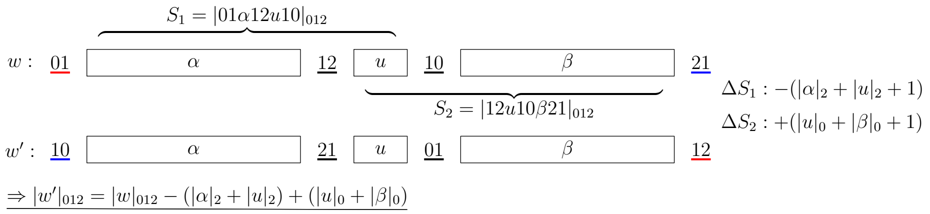
$S_1 = |01\alpha12u10|_{012}$
$w:$ 01 α 12 u 10 β 21
$S_2 = |12u10\beta21|_{012}$
$\Delta S_1 : -(|\alpha|_2 + |u|_2 + 1)$
$\Delta S_2 : +(|u|_0 + |\beta|_0 + 1)$
$w':$ 10 α 21 u 01 β 12
$\Rightarrow |w'|_{012} = |w|_{012} - (|\alpha|_2 + |u|_2) + (|u|_0 + |\beta|_0)$

Fig. 2. An illustration of substitutions for a word w such that $w = 01\alpha12u10\beta21$, where 12 occurs before 10.

While Figures 1 and 2 illustrate words with the same Parikh matrix by Proposition 5, there are other M-ambiguous words $w \in \Sigma_3^*$ that are not identified by Proposition 5, for instance, 012102021. Figure 3 depicts patterns of such words.

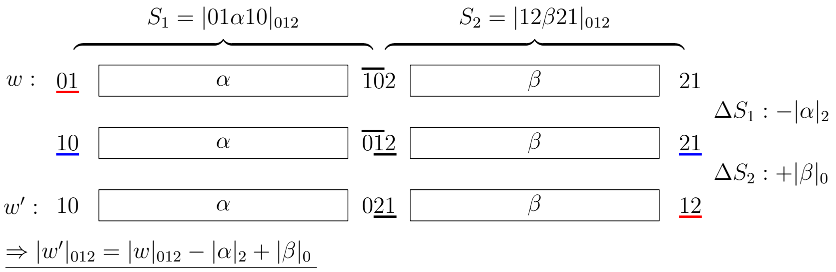
$S_1 = |01\alpha10|_{012}$
$S_2 = |12\beta21|_{012}$
$w:$ 01 α 102 β 21
$\Delta S_1 : -|\alpha|_2$
10 α 012 β 21
$\Delta S_2 : +|\beta|_0$
$w':$ 10 α 021 β 12
$\Rightarrow |w'|_{012} = |w|_{012} - |\alpha|_2 + |\beta|_0$

Fig. 3. An illustration of substitutions for a word w such that $w = 01\alpha102\beta21$ where w cannot maintain M-equivalence with a single substitution.

For M-equivalent words with such patterns, we develop substitution rules from Proposition 5 and introduce an equivalence relation of $(w, \Delta|w|_{012})$, the pair of a word w and a relative occurrence of 012.

Definition 5. *Given an ordered ternary alphabet $\Sigma_3 = \{0 < 1 < 2\}$, let $\cong$ be the minimal symmetric relation on $\Sigma_3^* \times \mathbb{Z}$ satisfying:*

R1. $(\alpha02\beta, k) \cong (\alpha20\beta, k)$,
R2. $(\alpha01u10\beta, k) \cong (\alpha10u01\beta, k - |u|_2)$ *for all u such that $|u|_2 \leq 1$, and*
R3. $(\alpha12u21\beta, k) \cong (\alpha21u12\beta, k + |u|_0)$ *for all u such that $|u|_0 \leq 1$,*

where $\alpha, \beta, u \in \Sigma_3^*$ and $k \in \mathbb{Z}$. Then, a $\cong$-sequence $(u_1, k_1), (u_2, k_2), \ldots, (u_n, k_n)$ is a sequence of pairs satisfying $(u_i, k_i) \cong (u_{i+1}, k_{i+1})$.

A relation $\cong^*$ is a minimal equivalence relation on $\Sigma_3^* \times \mathbb{Z}$ that satisfying:

R4. $(u, k) \cong (v, l)$ *implies* $(u, k) \cong^* (v, l)$, *and*
R5. $(\alpha u\beta, k) \cong^* (\alpha v\beta, l)$ *implies* $(xuy, k') \cong^* (xvy, l')$ *and* $k - l = k' - l'$,

where $\alpha, \beta, u, v, x, y \in \Sigma_3^*$ and $k, l, k', l' \in \mathbb{Z}$.

It is easy to verify that the minimal symmetric relation $\cong$ keeps the same values of $|u|_{01}$ and $|u|_{12}$ based on Proposition 4 and Definition 5. Note that, for binary words u and v over $\Sigma_2 = \{0 < 1\}$, we have $u \equiv_{pa}^* v$ if and only if $(u, k) \cong^* (v, k)$.

Proposition 6. *For any $u, v \in \Sigma_3^*$ and $k, l \in \mathbb{Z}$, $(u, k) \cong^* (v, l)$ implies $(u, k + c) \cong^* (v, l + c)$ for an arbitrary integer $c \in \mathbb{Z}$.*

We present Lemma 1 which generalizes R2 and R3 of Definition 5.

Lemma 1. *For any $u \in \Sigma_3^*$, $(01u10, k) \cong^* (10u01, k - |u|_2)$ and $(12u21, k) \cong^* (21u12, k + |u|_0)$.*

Proof. Consider the first case. For bases, if $|u|_2 \le 1$, then the relation holds by definition, if $|01u10| \le 5$, the relation holds.

Assume that, for all $N > 1$ and $L > 5$, $(01u10, k) \cong^* (10u01, k - |u|_2)$ for (1) $|u|_2 \le N$ and $|01u10| < L$ and (2) $|u|_2 < N$ and $|01u10| \le L$.

Consider $(01u10, 0)$ where $|u|_2 = N$, and $|01u10| = L$.

1. $|u|_1 = 0$. We show the congruence by prepending 21 and appending 12.

$$\begin{aligned}
&(2101u1012, 0) \\
\cong^*&(21012u'21012, 0) && [R1, (u, 0) \cong^* (2u'2, 0), u \in L((0+2)^*), |u|_2 \ge 2] \\
\cong^*&(12021u'12021, -2) && [\text{R3}] \\
\cong^*&(12201u'10221, -2) && [\text{R1}] \\
\cong^*&(12210u'01221, -2 - |u'|_2) && [\text{IH}; |01u'10| = L - 2 < L] \\
=&(12210u'01221, -|u|_2) && [|u'|_2 = |u|_2 - 2] \\
\cong^*&(21120u'02112, -|u|_2) && [\text{R3}] \\
\cong^*&(21102u'20112, -|u|_2) && [\text{R1}] \\
\cong^*&(2110u0112, -|u|_2) && [R1, (u, 0) \cong^* (2u'2, 0)]
\end{aligned}$$

2. $|u|_1 = 1$. If $u \in L((0+2)^*2(0+2)^*1(0+2)^*2(0+2)^*)$, we can use a procedure similar to that used in the case of $|u|_1 = 0$. Thus, let us assume that $u \in L((0+2)^*10^*)$ without loss of generality. By appending 01,

$$\begin{aligned}
&(01u1001, 0) \\
=&(01u'1v'1001, 0) && [u = u'1v', |u'|_1 = 0, |u'|_2 > 1, v' \in L(0^*)] \\
\cong^*&(01u'1v'0110, 0) && [\text{R2}] \\
=&(01u'10v'110, 0) \\
\cong^*&(10u'01v'110, -|u|_2) && [\text{IH}; |10u'01| = L - 1 < L.\ |u'|_2 = |u|_2] \\
\cong^*&(10u'10v'101, -|u|_2) && [\text{R2}] \\
=&(10u'1v'0101, -|u|_2) \\
=&(10u0101, -|u|_2)
\end{aligned}$$

3. $|u|_1 \geq 2$. We prepend 10 and append 01, then,

$$\begin{aligned}
&(1001u1001, 0) \\
\cong^*&(0110u0110, 0) && [\text{R2}] \\
=&(0110x1u'1y0110, 0) && [u = x1u'1y, u' \in \Sigma_3^*, x, y \in L((0+2)^*)] \\
\cong^*&(011x01u'10y110, 0) && [\text{R1}]
\end{aligned}$$

if $|x|_2 + |y|_2 > 0$, $|x|_2, |y|_2 \leq |u|_2$, then we can apply IH several times.

$$\begin{aligned}
&(011x01u'10y110, 0) \\
\cong^*&(011x10u'01y110, -|u'|_2) && [\text{IH}] \\
\cong^*&(101x01u'10y101, -|u|_2) && [\text{IH}] \\
=&(1010u0101, -|u|_2) && [\text{R1}]
\end{aligned}$$

Suppose not the case, i.e., $|x|_2 + |y|_2 = 0$. Then, by IH, $(01u'10, 0) \cong^* (10u'01, -|u|_2)$ holds; Note that $|01u'10| \leq L - 2 < L$ and $|u|_2 = |u'|_2$. Thus

$$\begin{aligned}
&(011x01u'10y110, 0) \\
\cong^*&(011x10u'01y110, -|u|_2) && [\text{IH, R5}] \\
\cong^*&(101x01u'10y101, -|u|_2) && [\text{R2}] \\
\cong^*&(1010x1u'1y0101, -|u|_2) && [\text{R1}] \\
=&(1010u0101, -|u|_2)
\end{aligned}$$

For every case, R5 implies that $(01u10, 0) \cong^* (10u01, -|u|_2)$. Thus, by induction, $(01u10, 0) \cong^* (10u01, -|u|_2)$ holds for all $u \in \Sigma_3^*$.

The second statement can be shown by swapping the roles of R2 and R3.

Lemma 2. *For two M-equivalent words u and v over a ternary alphabet $\Sigma_3 = \{0, 1, 2\}$, and two integers k and l such that $k \leq l$, let $S = [(u, k) \cong \cdots \cong (v, l)]$ be a $\cong$-sequence from (u, k) to (v, l). Then, for any integer t between k and l $(k \leq t \leq l)$, there exists a pair $(w, t) \in S$ for some $w \in \Sigma_3^*$.*

Proof. For the sake of contradiction, assume that there exists t such that $(u', t) \notin S$. Then, there must be two pairs $(u_i, t-1) \cong (u_{i+1}, t+1)$. However, by Definition 5, $|k_{i+1} - k_i| \leq 1$ for $(u_i, k_i) \cong (u_{i+1}, k_{i+1})$. This leads to a contradiction that such pairs of $(u_i, t-1)$ and $(u_{i+1}, t+1)$ do not exist. Therefore, the statement holds. □

Theorem 2. *Let Σ be an ordered ternary alphabet Σ_3. For two words $u, v \in \Sigma_3^*$ and two integers k, l, we have a $\cong$-sequence $S = [(u, k) \cong^* (v, l)]$ if and only if*

1. $|u|_{012} - |v|_{012} = k - l$,
2. $|u|_x = |v|_x$ for $x \in \{0, 1, 2, 01, 12\}$.

Proof. [only-if direction] We prove the statement by induction on the length of a $\cong$-sequence. For two words $u, v \in \Sigma_3^*$ and two integers k, l satisfying $(u, k) \cong^0$

(v,l) or $(u,k) \cong^1 (v,l)$, it is trivial to see that the two conditions hold. Note that, in the case of $\cong^0$, it is immediate that $u = v$ and $k = l$. Suppose that if $(u_i,k_i) \cong^i (v_i,l_i)$, then the two conditions hold for $2 \leq i < N$. For $(u,k) \cong^N (v,l)$, there exist two positive integers i, j such that $i+j = N$ and a pair (w,m) such that $(u,k) \cong^i (w,m) \cong^j (v,l)$. Thus, the statement holds for $\cong^*$.

From the statement, we know that if $(u_1,k_1) \cong^* (u_n,k_n)$, then there is a $\cong$-sequence whose length is bounded above by $\binom{|u|}{|u|_0} \cdot \binom{|u|-|u|_0}{|u|_1}$. In other words, there always exists a finite $\cong$-sequence.

[if direction] Since it is trivial when $u = v$, we assume that $u \neq v$. We prove by induction on the length of u and v. When $|u| = |v| \leq 3$, we prove the claim by checking every pair of words.

Our induction hypothesis (IH) is that, for $N \geq 4$, if we have two words u and v, which satisfy the two preconditions and $|u| = |v| < N$, then, we have $(u,k) \cong^* (v,l)$. Consider two words u and v of length N. When u and v share the same nonempty prefixes and suffixes, we can easily see that IH holds. Without loss of generality, let $u = au'$ and $v = av'$. We know that u' and v' satisfy the two preconditions and $(u',k) \cong^* (v',l)$. By R5 of Definition 5, $(u,k) \cong^* (v,l)$.

Thus for some symbols $u_p, u_s, v_p, v_s \in \Sigma_3$ such that $u_p \neq v_p$ and $u_s \neq v_s$, the strings u and v satisfies $u = u_p u' u_s$ and $v = v_p v' v_s$. In other words, they do not have common nonempty prefixes and suffixes.

It is easy to see that, if and only if u and v are in the following forms, $(u,k) \cong (v,l)$ holds:

1. $u = 01x10$, $v = 10x01$, or
2. $u = 21y12$, $v = 12y21$.

Note that $u = 02$ and $v = 20$ never occur due to the restriction on the length.

We now show that there exists a string x such that $S = [(u,k) \cong^* (x,c) \cong^* (v,l)]$, where $x = ax'b$ and $a, b \in \Sigma_3$. Then, only one of the following two cases holds:

1. All x's between (u,k) and (v,l) satisfies $u_p \neq a \neq v_p$ and $u_s \neq b \neq v_s$, or
2. at least one x between (u,k) and (v,l) satisfies $a \in \{u_p, v_p\}$ or $b \in \{u_s, v_s\}$.

For the first case, we subdivide $S = [(u,k) \cong (x_1,c_1) \cong^* (x_2,c_2) \cong (v,l)]$ into three subsequences $S_1 = [(u,k) \cong (x_1,c_1)]$, $S_2 = [(x_1,c_1) \cong^* (x_2,c_2)]$ and $S_3 = [(x_2,c_2) \cong (v,l)]$. It should be the case that $x_1 = ax'_1b$ and $x_2 = ax'_2b$, where $u_p \neq a \neq v_p$ and $u_s \neq b \neq v_s$. Then, the base cases cover S_1 and S_3, and we can apply R5 of Definition 5 on S_2. Note that all strings in S_2 have common prefixes and suffixes.

For the second case, we can subdivide S into two subsequences $S'_1 = [(u,k) \cong^* (x,c)]$ and $S'_2 = [(x,c) \cong^* (v,l)]$. Without loss of generality, let $a = u_p$. Then, u and x have a common prefix. We can detach the common prefix and IH applies on $u'u_s$ and $x'b$ thus the sequence S'_1 is covered. Note that Theorem 2 also applies on S'_2 because $|x|_{012} - |v|_{012} = (|u|_{012} - k + c) - |v|_{012} = (|u|_{012} - |v|_{012}) - k + c = (k - l) - k + c = c - l$ and the occurrences of length-1 and length-2 ordered scattered-factors are the same. Therefore, $(u,k) \cong^* (v,l)$. □

Theorem 2 provides a characterization for M-equivalence over a ternary alphabet. The following result is immediate from Theorem 2.

Corollary 1. *For a ternary alphabet Σ_3 and two words $u, v \in \Sigma_3^*$, $u \equiv_M v$ if and only if $(u, 0) \cong^* (v, 0)$.*

4 *M*-unambiguity

We investigate another property of the Parikh matrix, *M-unambiguity*. Recall that a word $w \in \Sigma^*$ is *M-unambiguous* if there is no word $w' \neq w$ such that $w \equiv_M w'$. Otherwise, w is *M-ambiguous*. Atanasiu et al. [3] established the family of M-ambiguous words over a binary alphabet. Then, Mateescu and Salomaa [8] first presented a regular expression of an M-unambiguous language over a binary alphabet.

Theorem 3. [8, Theorem 3] *For a binary alphabet $\Sigma_2 = \{0 < 1\}$, a word $w \in \Sigma_2^*$ is M-unambiguous if and only if*

$$w \in L(0^*1^* + 1^*0^* + 0^*10^* + 1^*01^* + 0^*101^* + 1^*010^*).$$

The regular expression in Theorem 3 is sufficient to identify M-unambiguous words over a binary alphabet. However, we cannot apply the same result to M-unambiguous words over a ternary alphabet. Şerbănuţă and Şerbănuţă [4] presented a collection of regular expressions of M-unambiguous words by enumerating all words for a ternary alphabet[2]. Based on Corollary 1, we establish an intuitive approach that computes a regular expression for M-ambiguous words and identifies M-unambiguous words.

Theorem 4. *Given a ternary alphabet $\Sigma_3 = \{0 < 1 < 2\}$, let $L \subseteq \Sigma_3^*$ be a regular language defined by the union of the following regular expressions.*

$$\begin{aligned}
E_1 &= \Sigma_3^* \cdot (02 + 01(0+1)^*10 + 10(0+1)^*01 + 12(1+2)^*21 + 21(1+2)^*12) \cdot \Sigma_3^* \\
E_2 &= \Sigma_3^* \cdot (01\Sigma_3^*2\Sigma_3^*10\Sigma_3^*10\Sigma_3^*2\Sigma_3^*01) \cdot \Sigma_3^* \\
E_3 &= \Sigma_3^* \cdot (01\Sigma_3^*2\Sigma_3^*10\Sigma_3^*12\Sigma_3^*0\Sigma_3^*21) \cdot \Sigma_3^* \\
E_4 &= \Sigma_3^* \cdot (21\Sigma_3^*0\Sigma_3^*12\Sigma_3^*10\Sigma_3^*2\Sigma_3^*01) \cdot \Sigma_3^* \\
E_5 &= \Sigma_3^* \cdot (21\Sigma_3^*0\Sigma_3^*12\Sigma_3^*12\Sigma_3^*0\Sigma_3^*21) \cdot \Sigma_3^* \\
E_6 &= \Sigma_3^* \cdot (01\Sigma_3^*12\Sigma_3^*10\Sigma_3^*21) \cdot \Sigma_3^* \\
E_7 &= \Sigma_3^* \cdot (10\Sigma_3^*21\Sigma_3^*01\Sigma_3^*12) \cdot \Sigma_3^*
\end{aligned}$$

and L^R be its reversal language $\{w^R \mid w \in L\}$.

Then, $L_{amb} = L \cup L^R$ is the set of all M-ambiguous words over Σ_3.

Proof. Let X be the set of all M-ambiguous words over Σ_3 and we show that $X = L_{amb}$. We prove that X is equivalent to L_{amb}.

[2] The regular expression is incorrect since it misses some M-unambiguous words illustrated in Figure 4 in Section 4.

$[X \subseteq L_{amb}]$. Suppose that there exists $u \in X \setminus L_{amb}$ and let $v \neq u$ be M-equivalent to u. Since $u \equiv_M v$, $(u, 0) \cong^* (v, 0)$ and thus, $(v, 0)$ is derived from $(u, 0)$ by a sequence of $\cong$ applications from Definition 5. For all the string patterns in Definition 5, we can easily find them in L_{amb}. For instance, E_2 contains $\alpha 01u10\beta$ as a prefix where $\alpha, \beta, u \in \Sigma_3$. Likewise, we can find the other string patterns of Definition 5. This contradicts that there exists u with distinct patterns that are not in L_{amb}. Therefore, $X \subseteq L_{amb}$.

$[L_{amb} \subseteq X]$. Suppose that there exists $u \in L_{amb} \setminus X$. This implies that u is M-unambiguous. Since $u \in L_{amb}$, we can derive v with s of Definition 5. We investigate when u is included in one of E_i of L_{amb}. When $u \in E_1$, we first examine when u contains 02 as a factor. By the first $\cong$-relation in Definition 5, u is M-ambiguous. There is also $u \in E_1$ that contains factors that are palindromicly amicable of a binary alphabet u is M-ambiguous by Theorem 1. Thus, $u \in E_1$ is M-ambiguous. Similarly, we can prove in the same way for the reversal of E_1.

For $u \in E_i$ for $2 \leq i \leq 7$, we inspect the change of 012 occurrence values by the second and the third $\cong$-relations of Definition 5. We show one of the cases when $u \in E_6$. When $u \in E_6$, the following holds for some $v \in \Sigma_3^*$:

$$\underbrace{(u, k) \cong^* (u', k - |\alpha 12\beta|_2)}_{01\alpha 12\beta 10 \to 10\alpha 12\beta 01} \overbrace{\cong^* (v, k - |\alpha 12\beta|_2 + |\beta 01\gamma|_0)}^{12\beta 01\gamma 21 \to 21\beta 01\gamma 12}.$$

Without loss of generality, let $0 < |\alpha 12\beta|_2 \leq |\beta 01\gamma|_0$. Then, $k - |\alpha 12\beta|_2 < k \leq k - |\alpha 12\beta|_2 + |\beta 01\gamma|_0$ and by Lemma 2, there exists (v', k) such that $(u, k) \cong^* (v', k) \cong^* (v, l)$ and $u \neq v'$. This leads to a contradiction that u is M-unambiguous because $(u, k) \cong^* (v', k)$ implies that v' is M-equivalent to u. We can prove similarly for E_2, E_3, E_4, E_5, E_7. Thus, $L_{amb} \subseteq X$. □

Theorem 4 establishes an identification for M-ambiguous words over a ternary alphabet. Then, the following result is immediate.

Corollary 2. *For $\Sigma_3 = \{0 < 1 < 2\}$ and $u \in \Sigma_3^*$, we have that u is M-unambiguous if and only if $u \notin L_{amb}$.*

Using the regular expression in Theorem 4, we find all M-unambiguous words that are missing in Şerbănuţă and Şerbănuţă [4]. Figure 4 is the minimal DFA for such missing M-unambiguous words.

5 Conclusions

We have presented a polished and complete characterization of M-equivalence and M-unambiguity over a ternary alphabet using $\cong^*$-relation. While the problem was solved for a binary alphabet, the larger-alphabet case has been open. We have presented key characteristics of M-equivalence and M-unambiguity over a ternary alphabet based on our substitution rules and $\cong^*$-relation. This result facilitates exploring further combinatorial properties of M-equivalent words.

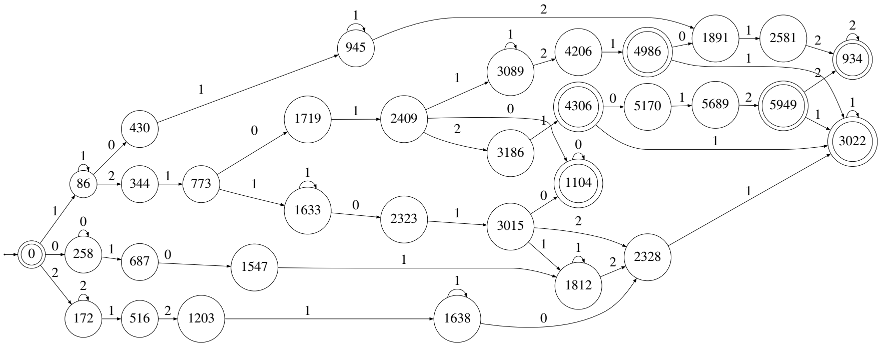

Fig. 4. An FA for M-unambiguous words missing in Şerbănuţă and Şerbănuţă [4].

Our equivalence relation is well-defined for a ternary alphabet but it can also be developed with further substitution rules for larger alphabets. We plan to extend $\cong$-relation to arbitrary alphabets and continue working towards establishing equivalent relations to M-equivalence and M-unambiguity. We also aim to address open problems related to other properties of Parikh matrices, such as ME-equivalence, strong M-equivalence, and weak M-relation [11, 14].

Acknowledgments

We thank all the reviewers for their valuable comments. We also appreciate Wen Chean Teh for pointing out an error in the previous manuscript.

This research was supported by the NRF grant funded by MIST (NRF-RS-2023-00208094). The first two authors contributed equally.

References

1. Atanasiu, A.: Parikh matrix mapping and amiability over a ternary alphabet. In: Discrete Mathematics and Computer Science. pp. 1–12 (2014)
2. Atanasiu, A., Atanasiu, R., Petre, I.: Parikh matrices and amiable words. Theoretical Computer Science **390**(1), 102–109 (2008)
3. Atanasiu, A., Martín-Vide, C., Mateescu, A.: On the injectivity of the Parikh matrix mapping. Fundamenta Informaticae **49**(4), 289–299 (2002)
4. Şerbănuţă, V.N., Şerbănuţă, T.F.: Injectivity of the Parikh matrix mappings revisited. Fundamenta Informaticae **73**(1–2), 265–283 (2006)
5. Fossé, S., Richomme, G.: Some characterizations of Parikh matrix equivalent binary words. Information Processing Letters **92**(2), 77–82 (2004)
6. Ibarra, O.H., Ravikumar, B.: On the Parikh membership problem for FAs, PDAs, and CMs. In: Proceedings of the 8th Language and Automata Theory and Applications. pp. 14–31 (2014)
7. Karhumäki, J.: Generalized Parikh mappings and homomorphisms. Information and Control **47**, 155–165 (1980)

8. Mateescu, A., Salomaa, A.: Matrix indicators for subword occurrences and ambiguity. International Journal of Foundations of Computer Science **15**(2), 277–292 (2004)
9. Mateescu, A., Salomaa, A., Salomaa, K., Yu, S.: A sharpening of the Parikh mapping. RAIRO Informatique Theorique et Applications **35**(6), 551–564 (2001)
10. Parikh, R.: On context-free languages. Journal of the ACM **13**(4), 570–581 (1966)
11. Poovanandran, G., Teh, W.C.: On M-equivalence and strong M-equivalence for Parikh matrices. International Journal of Foundations of Computer Science **29**(1), 123–138 (2018)
12. Salomaa, A.: On the injectivity of Parikh matrix mappings. Fundamenta Informaticae **64**(1–4), 391–404 (2005)
13. Teh, W.C.: On core words and the Parikh matrix mapping. International Journal of Foundations of Computer Science **26**(1), 123–142 (2015)
14. Teh, W.C., Subramanian, K.G., Bera, S.: Order of weak M-relation and Parikh matrices. Theoretical Computer Science **743**, 83–92 (2018)