On M-ambiguity of Words corresponding to a Parikh Matrix

Amrita Bhattacharjee
Department of Secondary Education
Silchar, Assam, India

ABSTRACT
M-ambiguous words are the problem of Parikh matrix. In this paper an algorithm is introduced to find the M-ambiguous ternary words corresponding to a 4x4 matrix. The concept of M-ambiguity Reduction factor is introduced. With the help of this M-ambiguity Reduction factor the problem of M-ambiguity can be solved to some extent.

Keywords
Parikh matrix, sub word, amiable words or M- ambiguous words.

1. INTRODUCTION
In 1966 R.J.Parikh introduced a notion called Parikh Mapping [13]. This notion is an important tool in the theory of formal languages. Using this tool words can be expressed as vectors. If the domain of a Parikh mapping is a context-free language then the range is always a semilinear set. It is a fact that from the transition of a sequence to a vector much of the information is lost about the sequence. To overcome this difficulty a new type of function based on matrix is introduced by [9]. This matrix based function was named as Parikh matrix. Parikh matrix of a word gives much more information about a word than Parikh vector. But Parikh Matrix faces some challenging problems, such as it is not injective. Two words may have the same corresponding Parikh Matrix. This problem is known as M-ambiguity and the words corresponding to a Parikh matrix is known as M-ambiguous words. To overcome all the shortcomings, Parikh Matrix has become a research interest in related fields in recent years. In recent decades many techniques have been developed to solve complex problems of words using Parikh Matrix. It is cited a few examples [8,10-12,14-17,19-20] which have used subword occurrences and Parikh matrix for solving the problems of word. The problem of M-ambiguity is dealt in the following papers [1-3, 18]. The papers [4-7] are also on various investigations about Parikh matrix.

Rest of this paper is organized as follows. The following section goes toward the basic ideas about Parikh matrix. Third section gives algorithm which gives ternary words, (may be one word or more than one word due to M-ambiguity) corresponding to a Parikh matrix; In fourth section notion of M-ambiguity Reduction factor is introduced. In the fifth section the paper is concluded.

2. PRELIMINARIES
Throughout this paper Z will denote the set of natural numbers including zero. Some necessary definitions are given below.

Ordered alphabet: an ordered alphabet is a set of symbols $\Sigma = \{a_1, a_2, a_3, \cdots, a_n\}$ where the symbols are arranged maintaining a relation of order ("$<$") on it. For example if $a_1 < a_2 < a_3 < \cdots < a_n$, then we use notation: $\Sigma = \{a_1, a_2, a_3, \cdots, a_n\}$

Word: a word is a finite or infinite sequence of symbols taken from a finite set called an alphabet. Let $\Sigma = \{a_1, a_2, a_3, \cdots, a_n\}$ be the alphabet. The set of all words over Σ is Σ^*. The empty word is denoted by λ. $|w|_{a_i}$: Let $a_i \in \Sigma = \{a_1, a_2, a_3, \cdots, a_n\}$ be a letter. The number of occurrences of a_i in a word $w \in \Sigma^*$ is denoted by $|w|_{a_i}$.

Sub -word: a word u is a sub- word of a word w, if there exists words $x_1 \cdots x_n$ and $y_0 \cdots y_n$, (some of them possibly empty), such that $u = x_1 \cdots x_n$ and $w = y_0 x_1 y_1 \cdots x_n y_n$. For example if $w = adbaabcacd$ is a word over the alphabet $\Sigma = \{a, b, c, d\}$ then $baca$ is a sub-word of w. Two occurrences of a sub-word are considered different if they differed by at least one position of some letter. In the word $w = adbaabcacd$, the number of occurrences of the word $baca$ as a sub-word of w is $|w|_{baca} = 2$.

Parikh Vector: The Parikh vector is a mapping $\Psi : \Sigma^* \to Z \times Z \times \cdots \times Z$ where $\Sigma = \{a_1, a_2, a_3, \cdots, a_n\}$ and Z is the set of natural numbers including 0, such that for a word w in Σ^*, $\Psi(w) = (|w|_{a_1}, |w|_{a_2}, |w|_{a_3}, \cdots, |w|_{a_n})$ with $|w|_{a_i}$ denoting the number of occurrences of the letter $a_i \in w$. For example, for the word $w = abaabcac$ the Parikh vector is (4, 2, 2).

Triangle matrix: A triangle matrix is a square matrix $m = (m_{ij})_{1 \le i, j \le n}$ such that:

1. $m_{ij} \in Z \quad (1 \le i, j \le n)$,
2. $m_{ij} = 0$ for all $1 \le j < i \le n$,
3. $m_{ii} = 1 \quad (1 \le i \le n)$.

Parikh matrix: let $\Sigma = \{a_1 < a_2 < a_3 < \cdots < a_n\}$ be an ordered alphabet, where $n \geq 1$.The Parikh matrix mapping, denoted Ψ_{M_n}, is the homomorphism $\Psi_{M_n} : \Sigma^* \to M_{n+1}$ defined as follows: if $\Psi_{M_n}(a_q) = (m_{i\,j})_{1 \leq i,\, j \leq n+1}$ then $m_{i,i} = 1,\ m_{q,q+1} = 1$ and all other elements of the matrix $\Psi_{M_n}(a_q)$ are zero.

M-ambiguous words: Two words $\alpha, \beta \in \sum^*, (\alpha \neq \beta)$ over the same alphabet Σ may have the same Parikh matrix. Then the words are called amiable or M-ambiguous. The words *baaabaa* and *ababaaa* have the same Parikh matrix. $\begin{pmatrix} 1 & 5 & 3 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{pmatrix}$ So these two words are amiable.

M-unambiguous words: A word *w* is said to be M-unambiguous if there is no word w' for which $\Psi_{M_n}(w) \neq \Psi_{M_n}(w')$

3. ALGORITHM FOR DISPLAYING M-AMBIGUOUS WORDS CORRESPONDING TO A PARIKH MATRIX OF A TERNARY SEQUENCE

3.1 Algorithm giving Parikh matrix corresponding to a ternary sequence

The following pseudo code gives the Parikh matrix of a ternary sequence [7].

Algorithm:

1. Initialize a word = '*w*'
2. Set *len* = length of *w*
3. **For** *i=0* to *len* do
4. Calculate total number of *a*, *ab*, *abc* in *w*.
5. Calculate total number of *b*, *bc* in *w*.
6. Calculate total number of *c* in *w*.
7. **End**.
   // create a matrix (a_{ij}) of order *M (=4)*
8. **For** *i=0* to *M* do
9. **For** *j=0* to *M* do
10. **If** *i=j*
11. a_{ij}=1
12. else If **(***i* > *j***)**
13. a_{ij} *=0*
14. else
15. **If** *(i=0 and j=1)*
16. a_{ij}=total number of '*a*'
17. **If***(i=0 and j=2)*
18. a_{ij}=total number of '*ab*'
19. **If** *(i=0 and j=3)*
20. a_{ij}=total number of '*abc*'
21. **If** *(i=1 and j=2)*
22. a_{ij}=total number of '*b*'
23. **If** *(i=1 and j=3)*
24. a_{ij}=total number of '*bc*'
25. **If** *(i=2 and j=3)*
26. a_{ij}=total number of 'c'
27. **End**
28. **End**

3.2 Algorithm giving ternary sequences corresponding to a Parikh matrix

Various methods are used for finding M-ambiguous words for binary words [6]. The algorithm given below is introduced to find M-ambiguous words for ternary words. With the help of this algorithm all the M-ambiguous words corresponding to a 4x4 Parikh matrix can be found instantly. One just has to enter a 4x4 Parikh matrix. If the matrix is not a Parikh matrix then there will be simply no corresponding word.

Algorithm:

1. Input a matrix $A_{i,j}$ of order 4x4.
2. $l = (A_{0,1} + A_{1,2} + A_{2,3})$
3. Store '*l*' time of '*abc*' in *w* and consider it as an initial word $A_{i,j}$
4. Make a list of words *L* where each word is obtained from *w*. (*w* is also included in *L*)
5. *L*= {$l_1, l_2, l_3, \ldots, l_{length\ of\ w}$ }, l_i = word obtained from *w* by removing the i^{th} letter from it.
6. *len* = Number of elements of (*L*)
7. **For** *i=1* to *len* do
8. *W* = word at i^{th} position of *L*
9. **For** *j=1* to length of *W* do
10. Store *W* by removing character at j^{th} position
11. **End**
12. Remove all duplicate copies of words from *L*
13. Remove *W* from *L* if numbers of *a* in W ≠ $A_{0,1}$ numbers of *b* in W ≠ $A_{1,2}$ and numbers of *c* in W ≠ $A_{2,3}$
14. If it is removed then update *i* by *i-1*
15. Update *len* = Number of elements of (*L*)
16. If all the words of *L* is of length *l* display the words
17. **End**
18. **For** *k=1* to *len* do
19. Perform Algorithm of 3.1 with each word at k^{th} position
20. It gives matrix X_{ij} of size 4 x 4
21. Check each of X_{ij} with $A_{i,j}$
22. If it matches then display 'YES' otherwise display 'NO'.
23. **End**.

3.3 Application of the algorithm giving ternary sequences corresponding to a Parikh matrix

(a) Let $\Sigma = \{a < b < c\}$ and the Parikh matrix be

$$\Psi_{M_3}(\zeta_1) = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 2 & 2 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

the set of amiable words having this Parikh matrix is $C = \{cbbac, cbbca, bccba\}$

(b) Let $\Sigma = \{a < b < c\}$ and the Parikh matrix be

$$\Psi_{M_3}(\zeta_2) = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

the set of amiable words having this Parikh matrix is

$C = \{bcbac, bcbca\}$

(c) Let $\Sigma = \{a < b < c\}$ and the Parikh matrix be

$$\Psi_{M_3}(\zeta_3) = \begin{pmatrix} 1 & 2 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

the set of amiable words having this Parikh matrix is

$C = \{abac, abca\}$

4. SOLVING M-AMBIGUITY USING REDUCTION FACTOR

4.1 Representation of M- ambiguous words by Parikh matrix together with an M-ambiguity reduction factor

For any word ζ in ternary sequence one can represent each word as repetition of $a^{x_i}b^{y_i}c^{z_i}$ where x_i = number of *a*'s placed together in the word without intercepted by *b*'s and *c*'s. y_i = number of *b*'s placed together without intercepted by *c*'s and *a*'s, z_i = number of *c*'s placed together without intercepted by *a*'s and *b*'s. So, the representation of a word ζ is $\zeta = a^{x_1}b^{y_1}c^{z_1}a^{x_2}b^{y_2}c^{z_2}\cdots a^{x_n}b^{y_n}c^{z_n}$. The value of *i* in this type of representation is determined by breaking of the sequence *a*'s < *b*'s < *c*'s < in the word. For example, let us take a ternary word *aabbbacc*. By the above process this word can be written as $a^2b^3c^0a^1b^0c^2$. After representing the word by above method the value of the function $R(\zeta) = \sqrt{x_1^2 + y_1^2 + z_1^2} \cdot \sqrt{x_2^2 + y_2^2 + z_2^2} \cdots \sqrt{x_n^2 + y_n^2 + z_n^2}$ is calculated. The value of *R (ζ)* for the above word *aabbbacc* is as follows:

$$R(aabbbacc) = R(a^2b^3c^0a^1b^0c^2)$$

This calculated value *R (ζ)* is not always the same for the M-ambiguous words corresponding to a Parikh matrix. Some M-ambiguous words can have the same value of the function *R(ζ)* but some of them can have a unique value. So those words for which *R(ζ)* take a unique value can be sorted out and can be uniquely represented by the Parikh matrix and the unique value of *R(ζ)*. This function *R(ζ)* is named as M-ambiguity reduction factor. Along with the Parikh matrix if we use this number for every word then the problem of M-ambiguity can be handled to a great extent.

4.2 Example of solving M-ambiguity using M-ambiguity reduction factor for binary sequence

Let us consider the M-ambiguous words for the following matrix $\Psi_{M_2}(\xi) = \begin{pmatrix} 1 & 5 & 9 \\ 0 & 1 & 5 \\ 0 & 0 & 1 \end{pmatrix}$. The set of amiable words having this Parikh matrix is

$$C = \{bbaabababa, bbabaabaab, babbaababa, babbabaaab, \\ bababbaaba, babababbaa, abbbabaaba, abbabbabaa, \\ bbbaaaabab, bbaabbaaab, bbabaaabba, baabbbabaa, \\ abbbbaaaab, abbbaabbaa, ababbbbaaa, bbaaabbbaa\}$$

Let us find the respective numbers corresponding to the sixteen M-ambiguous words one by one.

$\xi_1 = bbaabababa = a^0b^2a^2b^1a^1b^1a^1b^1a^1b^0$

$\therefore R(\xi_1) = \sqrt{0+2^2}\sqrt{2^2+1}\sqrt{1+1}\sqrt{1+1}\sqrt{1+0} = 4\sqrt{5}$

$\xi_2 = bbabaabaab = a^0b^2a^1b^1a^2b^1a^2b^1$

$\therefore R(\xi_2) = \sqrt{0+2^2}\sqrt{1+1}\sqrt{2^2+1}\sqrt{2^2+1} = 10\sqrt{2}$

$\xi_3 = babbaababa = a^0b^1a^1b^2a^2b^1a^1b^1a^1b^0$

$\therefore R(\xi_3) = \sqrt{0+1}\sqrt{1+2^2}\sqrt{2^2+1}\sqrt{1+1}\sqrt{1+0} = 5\sqrt{2}$

$\xi_4 = babbabaaab = a^0b^1a^1b^2a^1b^1a^3b^1$

$\therefore R(\xi_4) = \sqrt{0+1}\sqrt{1+2^2}\sqrt{1+1}\sqrt{3^2+1} = 10$

$\xi_5 = bababbaaba = a^0b^1a^1b^1a^1b^2a^2b^1a^1b^0$

$\therefore R(\xi_5) = \sqrt{0+1}\sqrt{1+1}\sqrt{1+2^2}\sqrt{2^2+1}\sqrt{1+0} = 5\sqrt{2}$

$\xi_6 = babababbaa = a^0b^1a^1b^1a^1b^1a^1b^2a^2b^0$

$\therefore R(\xi_6) = \sqrt{0+1}\sqrt{1+1}\sqrt{1+1}\sqrt{1+2^2}\sqrt{2^2+0} = 4\sqrt{5}$

$\xi_7 = abbbabaaba = a^1b^3a^1b^1a^2b^1a^1b^0$

$\therefore R(\xi_7) = \sqrt{1+3^2}\sqrt{1+1}\sqrt{2^2+1}\sqrt{1+0} = 10$

$\xi_8 = abbabbabaa = a^1b^2a^1b^2a^1b^1a^2b^0$

$\therefore R(\xi_8) = \sqrt{1+2^2}\sqrt{1+2^2}\sqrt{1+1}\sqrt{2^2+0} = 10\sqrt{2}$

$\xi_9 = bbbaaaabab = a^0b^3a^4b^1a^1b^1$

$\therefore R(\xi_9) = \sqrt{0+3^2}\sqrt{4^2+1}\sqrt{1+1} = 3\sqrt{34}$

$\xi_{10} = bbaabbaaab = a^0b^2a^2b^2a^3b^1$

$\therefore R(\xi_{10}) = \sqrt{0+2^2}\sqrt{2^2+2^2}\sqrt{3^2+1} = 8\sqrt{5}$

$\xi_{11} = bbabaaabba = a^0b^2a^1b^1a^3b^2a^1b^0$

$\therefore R(\xi_{11}) = \sqrt{0+2^2}\sqrt{1+1}\sqrt{3^2+2^2}\sqrt{1+0} = 2\sqrt{26}$

$\xi_{12} = baabbbabaa = a^0b^1a^2b^3a^1b^1a^2b^0$

$\therefore R(\xi_{12}) = \sqrt{0+1}\sqrt{2^2+3^2}\sqrt{1+1}\sqrt{2^2+0} = 2\sqrt{26}$

$\xi_{13} = abbbbaaaab = a^1b^4a^4b^1$

$\therefore R(\xi_{13}) = \sqrt{1+4^2}\sqrt{4^2+1} = 17$

$\xi_{14} = abbbaabbaa = a^1b^3a^2b^2a^2b^0$

$\therefore R(\xi_{14}) = \sqrt{1+3^2}\sqrt{2^2+2^2}\sqrt{2^2+0} = 8\sqrt{5}$

$\xi_{15} = ababbbbaaa = a^1b^1a^1b^4a^3b^0$

$\therefore R(\xi_{15}) = \sqrt{1+1}\sqrt{1+4^2}\sqrt{3^2+0} = 3\sqrt{34}$

$\xi_{16} = bbaaabbbaa = a^0b^2a^3b^3a^2b^0$

$\therefore R(\xi_{16}) = \sqrt{0+2^2}\sqrt{3^2+3^2}\sqrt{2^2+0} = 12\sqrt{2}$

Among these sixteen M-ambiguous words two words namely ξ_{13} and ξ_{16} can be represented uniquely by Parikh matrix along with the reduction factor $R(\xi_i)$

4.3 Examples of solving M-ambiguity using M-ambiguity reduction factor for ternary sequence

Let us consider the M-ambiguous words for the following matrix. Let $\Sigma = \{a < b < c\}$ and the Parikh matrix be $\Psi_{M_3}(\zeta) = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 2 & 2 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$, the set of amiable words having this matrix is $C = \{cbbac, cbbca, bccba\}$. Let us find the respective numbers corresponding to the three M-ambiguous words one by one

$$\zeta_1 = cbbac = a^0b^0c^1a^0b^2c^0a^1b^0c^1$$

$$\therefore R(\zeta_1) = \sqrt{0+0+1}\sqrt{0+2^2+0}\sqrt{1+0+1} = 2\sqrt{2}$$

$$\zeta_2 = cbbca = a^0b^0c^1a^0b^2c^1a^1b^0c^0$$

$$\therefore R(\zeta_2) = \sqrt{0+0+1}\sqrt{0+2^2+1}\sqrt{1+0+0} = \sqrt{5}$$

$$\zeta_3 = bccba = a^0b^1c^2a^0b^1c^0a^1b^0c^0$$

$$\therefore R(\zeta_3) = \sqrt{0+1+2^2}\sqrt{0+1+0}\sqrt{1+0+0} = \sqrt{5}$$

Among these M-ambiguous words one word ζ_1 can be represented uniquely by Parikh matrix along with the reduction factor $R(\zeta_i)$. Again let us consider the M-ambiguous words for the matrix

$$\Psi_{M_3}(\alpha) = \begin{pmatrix} 1 & 2 & 4 & 4 \\ 0 & 1 & 3 & 4 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

There are seven M-ambiguous words corresponding to the above matrix and these are

$$\alpha_1 = abbaccb, \alpha_2 = abbcacb, \alpha_3 = abbccab$$

$$\alpha_4 = baabccb, \alpha_5 = baacbbc, \alpha_6 = bacabbc$$

$$\alpha_7 = bcaabbc$$

Let us find the respective numbers corresponding to the seven M-ambiguous words one by one.

$$\alpha_1 = abbaccb \rightarrow a^1b^2c^0a^1b^0c^2a^0b^1c^0$$

$$\therefore R(\alpha_1) = \sqrt{1+2^2+0}\sqrt{1+0+2^2}\sqrt{0+1+0} = 5$$

$$\alpha_2 = abbcacb \rightarrow a^1b^2c^1a^1b^0c^1a^0b^1c^0$$

$$\therefore R(\alpha_2) = \sqrt{1+2^2+1}\sqrt{1+0+1}\sqrt{0+1+0} = 2\sqrt{3}$$

$$\alpha_3 = abbccab \rightarrow a^1b^2c^2a^1b^1c^0$$

$$\therefore R(\alpha_3) = \sqrt{1+2^2+2^2}\sqrt{1+1+0} = 3\sqrt{2}$$

$$\alpha_4 = baabccb \rightarrow a^0b^1c^0a^1b^0c^0a^1b^1c^2a^0b^1c^0$$

$$\therefore R(\alpha_4) = \sqrt{0+1+0}\sqrt{1+0+0}\sqrt{1+1+2^2}\sqrt{0+1+0} = \sqrt{6}$$

$$\alpha_5 = baacbbc \rightarrow a^0b^1c^0a^1b^0c^0a^1b^0c^1a^0b^2c^1$$

$$\therefore R(\alpha_5) = \sqrt{0+1+0}\sqrt{1+0+0}\sqrt{1+0+1}\sqrt{0+2^2+1} = \sqrt{2}\sqrt{5}$$

$$\alpha_6 = bacabbc \rightarrow a^0b^1c^0a^1b^0c^1a^1b^2c^1$$

$$\therefore R(\alpha_6) = \sqrt{0+1+0}\sqrt{1+0+1}\sqrt{1+2^2+1} = 2\sqrt{3}$$

$$\alpha_7 = bcaabbc \rightarrow a^0b^1c^1a^1b^0c^0a^1b^2c^1$$

$$\therefore R(\alpha_7) = \sqrt{0+1+1}\sqrt{1+0+0}\sqrt{1+2^2+1} = 2\sqrt{3}$$

Among these seven M-ambiguous words four words namely $\alpha_1, \alpha_3, \alpha_4, \alpha_5$ can be represented uniquely by Parikh matrix along with the reduction factor $R(\alpha_i)$.

4.4 Example of solving M-ambiguity using M-ambiguity reduction factor for tertiary sequence

The following two words $\beta_1 = abcddcbadcba$ and $\beta_2 = abdccbddacba$ are M-ambiguous words corresponding to the Parikh matrix $\Psi_{M_4}(\beta) = \begin{pmatrix} 1 & 3 & 4 & 4 & 4 \\ 0 & 1 & 3 & 4 & 4 \\ 0 & 0 & 1 & 3 & 4 \\ 0 & 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$. Let us find the respective numbers corresponding to the two M-ambiguous words one by one.

$$\beta_1 = abcddcbadcba$$

$$= a^1b^1c^1d^2a^0b^0c^1d^0a^0b^1c^0d^0a^1b^0c^0d^1 a^0b^0c^1d^0a^0b^1c^0d^0a^1b^0c^0d^0$$

$$\therefore R(\beta_1) = \sqrt{1+1+1+2^2}\sqrt{0+0+1+0}\sqrt{0+1+0+0} \sqrt{1+0+0+1}\sqrt{0+0+1+0}\sqrt{0+1+0+0} \sqrt{1+0+0+0} = \sqrt{14}$$

$$\beta_2 = abdccbddacba$$

$$= a^1b^1c^0d^1a^0b^0c^2d^0a^0b^1c^0d^2 a^1b^0c^1d^0a^0b^1c^0d^0a^1b^0c^0d^0$$

$$\therefore R(\beta_2) = \sqrt{1+1+0+1}\sqrt{0+0+2^2+0}\sqrt{0+1+0+2^2} \sqrt{1+0+1+0}\sqrt{0+1+0+0}\sqrt{1+0+0+0} = 2\sqrt{30}$$

All these M-ambiguous words can be represented uniquely by Parikh matrix along with the reduction factor $R(\beta_i)$

5. CONCLUSION

Parikh matrix helps in the numerical representation of a word. The representation is not unique due to M-ambiguity. In this paper an algorithm is developed, which is meant for finding all the M- ambiguous words from a ternary Parikh matrix. To overcome the problem of M-ambiguity a function M-ambiguity Reduction factor is introduced. This function can solve the M-ambiguity property of Parikh matrix to some extent.

6. REFERENCES

[1] Atanasiu, A. 2007. Binary amiable words, Int. J. Found. Comput. Sci 18(2), 387- 400.

[2] Atanasiu, A., Atanasiu, R and Petre, I. 2008. Parikh matrices and amiable words, Theoretical Computer Science 390, 102--109.

[3] Atanasiu, A., Vide C.M. and Mateescu, A. 2001. On the injectivity of the parikh matrix mapping, Fundam. Informa. 46,1-11.

[4] Bhattacharjee, A and Purkayastha, B.S. 2014. Parikh matrices and words over tertiary ordered alphabet, International Journal of Computer Applications 85(4), 10-15.

[5] Bhattacharjee, A and Purkayastha, B.S. 2014.Application of ratio property in searching of m-ambiguous words and its generalization, 3rd International Conference on Soft Computing for Problem Solving, AISC (SocProS 2013), Springer India 258, 857-865.

[6] Bhattacharjee, A and Purkayastha, B.S. 2014. Some alternative ways to find M-ambiguous binary words corresponding to a Parikh matrix, International Journal on Computational Sciences and Applications (IJCSA) 4(1),53-64.

[7] Bhattacharjee, A and Purkayastha, B.S. 2014. Parikh matrices and words over ternary alphabet, 4th International Conference on Soft Computing for Problem Solving, AISC (SocProS 2014), Springer India 335,135-145.

[8] Ding, C and Salomaa, A. 2006. On some problems of Mateescu concerning sub word occurences, Fundamenta Informaticae 72, 1-15.

[9] Mateescu, A., Salomaa, A., Salomaa, K and Yu, S. 2001 A sharpening of the parikh mapping, Theoret. Informetics Appl. 35,551-564.

[10] Mateescu, A., Salomaa, A., Salomaa, K and Yu, S. 2001. On an extension of the Parikh mapping, T.U.C.S Technical Report No 364.

[11] Mateescu, A., Salomaa, A and Yu, S. 2004. Subword histories and parikh matrices, J. Comput. Syst. Sci. 68, 1-21.

[12] Mateescu, A and Salomaa, A. 2004. Matrix indicators for subword occurences and ambiguity, Int. J. Found. Comput. Sci. 15, 277-292.

[13] Parikh, R.J. 1966. On the context-free languages, Journal of the Association for Computing Machinery 13, 570-581.

[14] Salomaa, A. 2003. Counting (scattered) subwords, EATCS Bulletin 81, 165-179.

[15] Salomaa, A. 2005. Connections between subwords and certain matrix mappings, Theoretical Computer Science 340,188-203.

[16] Salomaa, A. 2006. Independence of certain quantities indicating subword occurrences, Theoretical Computer Science 362(1), 222-231.

[17] Salomaa, A. et. al, 2006. Subword conditions and subword histories, Information and Computation 204, 1741-1755.

[18] Serbanuta, V.N. and Serbanuta. T.F. 2006. Injectivity of the Parikh matrix mappings revisited, Fundamenta Informaticae, XX, IOS Press, 1-19.

[19] Subramanian, K.G., Huey, A. M and Nagar, A. K. 2009. On parikh matrices, Int. J. Found. Comput. Sci. 20(2), 211-219.

[20] Subramanian, K.G., Isawasan, P and Venkat, I. 2013.Parikh matrices and istrail morphism, Malaysian Journal of Fundamental and Applied Sciences 9(1), 5-9.