ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 1, January 2014

# A Novel Approach for Encryption of Text Messages Using PLAY-FAIR Cipher 6 by 6 Matrix with Four Iteration Steps

Nisarga Chand, Subhajit Bhattacharyya

***Abstract*— Play fair Cipher algorithm has demonstrated its success for encryption of text messages over the years. This algorithm uses normally 5 by 5 matrices.[5] The main disadvantages of this matrix is we cannot use numeric and also have to compromise between I and J. Here we use this concept but extend the matrix dimension 6 by 6, so that we can include numeric as well as I and J. To make the algorithm stronger here we use four iteration steps instead of one. Finally we have implemented this concept with the help of MATLAB.***

***Index Terms* — Playfair cipher, Encryption, Security, Ad-hoc network.**

## I. INTRODUCTION

This paper is a step toward developing an encryption system which can encrypt any text message securely. An ad-hoc network generally consists of nodes, on which sensors are embedded to provide security measures. The main challenge of these sensors is to provide security of data and also to work effectively within a limitation of power and memory [1]. In every important sector these networks are used to collect information or to transfer them with a high level of security. For this reason here we require a strong encryption technique. Here we choose play fair cipher algorithm which is very strong and also require less memory and power [2]. The play fair cipher or play fair square is a manual symmetric encryption technique and was the first literal digraph substitution cipher. The scheme was invented in 1854 by Charles Wheatstone, but bears the name of Lord Play fair who promoted the use of the cipher [4]. The technique encrypts pairs of letters (digraphs), instead of single letters as in the simple substitution cipher. The Play fair cipher uses a 5 by 5 table containing a key word or phrase. Memorization of the keyword and 4 simple rules was all that was required to create the 5 by 5 table and use the cipher. There are two main disadvantages of this traditional play fair cipher matrix. First one is we have to compromise between I and J and the second one is we cannot include numeric values in this matrix. To overcome this problem we have developed a new concept which includes 6 by 6 play fair cipher matrix. This matrix consists of alphabets A to Z and numeric values 0 to 9. Here we use four iteration steps to make strong encrypted message. The organization of the paper can be summarized as: The traditional playfair cipher explained in Section-II. Extended 6 by 6 playfair cipher algorithm with flowchart explained in Section-III. Experimental results are shown in Section-IV. Future works are discussed in Section-V. Conclusions are explained in Section-VI.

## II. TRADITIONAL PLAY-FAIR CIPHER ALGORITHM

The Play fair cipher [7] uses a 5 by 5 table containing a key word or phrase. To generate the key table, one would first fill in the spaces in the table with the letters of the keyword (dropping any duplicate letters), then fill the remaining spaces with the rest of the letters of the alphabet in order (usually omitting "Q" to reduce the alphabet to fit, other versions put both "I" and "J" in the same space). The key can be written in the top rows of the table, from left to right, or in some other pattern, such as a spiral beginning in the upper-left-hand corner and ending in the center. The keyword together with the conventions for filling in the 5 by 5 table constitutes the cipher key. To encrypt a message, one would break the message into digraphs (groups of 2 letters) such that, for example, "Hello World" becomes “HE LL OW OR LD", and map them out on the key table. The two letters of the digraph are considered as the opposite corners of a rectangle in the key table. Note the relative position of the corners of this rectangle. Then apply the following 4 rules, in order, to each pair of letters in the plaintext:

i) If both letters are the same (or only one letter is left), add an "X" after the first letter. Encrypt the new pair and continue. Some variants of Play fair use "Q" instead of "X", but any uncommon monograph will do.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 1, January 2014

ii) If the letters appear on the same row of your table, replace t hem with the letters to their immediate right respectively (wrapping around to the left side of the row if a letter in the original pair was on the right side of the row).
iii) If the letters appear on the same column of your table, replace them with the letters immediately below respectively (wrapping around to the top side of the column if a letter in the original pair was on the bottom side of the column).
iv) If the letters are not on the same row or column, replace them with the letters on the same row respectively but at the other pair of corners of the rectangle defined by the original pair. The order is important – the first letter of the encrypted pair is the one that lies on the same row as the first letter of the plaintext pair.
To decrypt, use the INVERSE (opposite) of the first 3 rules, and the 4th as -is (dropping any extra "X"s (or "Q"s) that don't make sense in the final message when finished).
PROCEDURE: - Using "playfair example" as the key, Table becomes:

| | | | | |
|---|---|---|---|---|
| P | L | A | Y | F |
| I | R | E | X | M |
| B | C | D | G | H |
| K | N | O | Q | S |
| T | U | V | W | Z |

Here we shift the alphabets in such a way that no one can read the code word without knowing the procedure. It is totally for security purpose or prevent from hackers. So the steps are as follows:
Encrypting the message - “HIDE THE GOLD IN THE TREE STUMP": HI DE TH EG OL DI NT HE TR EX ES TU MP
1. The pair HI forms a rectangle, replace it with BM.
2. The pair DE is in a column replaces it with OD.
3. The pair TH forms a rectangle, replace it with ZB.
4. The pair EG forms a rectangle, replace it with XD.
5. The pair OL forms a rectangle, replace it with NA.
6. The pair DI forms a rectangle, replace it with BE.
7. The pair NT forms a rectangle, replace it with KU.
8. The pair HE forms a rectangle, replace it with DM.
9. The pair TR forms a rectangle, replace it with UI.
10. The pair EX (X inserted to split EE) is in a row, replace it with XM.
11. The pair ES forms a rectangle, replace it with MO.
12. The pair TU is in a row replaces it with UV.
13. The pair MP forms a rectangle, replace it with IF.
Thus the message “HIDE THE GOLD IN THE TREE STUMP” becomes
"BMODZBXDNABEKUDMUIXMMOUVIF".

## III. EXTENDED 6 BY 6 PLAYFAIR CIPHER ALGORITHM USING FOUR ITERATION STEPS

This extended play fair algorithm is based on the use of a 6 X 6 matrix of letters constructed using a keyword. The matrix is constructed by filling in the letters of the keyword from left to right and from top to bottom, and the filling in the remainder of the matrix with the remaining letters in alphabetic order and digits in ascending order form 0 to 9. The digits 0 to 9 can be placed next cells of the alphabet z in an ascending order. In this we have not counted I/J as one letter instead we are placing both I and J in two different cells in order to avoid the ambiguity to the user at the time of decipherment. This algorithm can allow the plain text containing of alpha numeric values; hence the user can easily encrypt alpha numeric values efficiently. The plain text containing contact numbers, date of birth, house numbers and other numerical values can be easily and efficiently encrypted using this algorithm.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 1, January 2014

### *A. Assumption*

Here we have used four reserved keywords: MONARCHY, PLAYFAIR, ENCRYPTION and DIAMONDS. Then we construct four 6 by 6 matrices with the help of these four keywords. The four 6 by 6 matrices are shown in figure 1-4.

**Keyword: MONARCHY**

| | | | | | |
|---|---|---|---|---|---|
| **M** | **O** | **N** | **A** | **R** | **C** |
| **H** | **Y** | B | D | E | F |
| G | I | J | K | L | P |
| Q | S | T | U | V | W |
| X | Z | 0 | 1 | 2 | 3 |
| 4 | 5 | 6 | 7 | 8 | 9 |

**Figure 1**

**Keyword: PLAYFAIR**

| | | | | | |
|---|---|---|---|---|---|
| **P** | **L** | **A** | **Y** | **F** | **I** |
| **R** | B | C | D | E | G |
| H | J | K | M | N | O |
| Q | S | T | U | V | W |
| X | Z | 0 | 1 | 2 | 3 |
| 4 | 5 | 6 | 7 | 8 | 9 |

**Figure 2**

**Keyword: ENCRYPTION**

| | | | | | |
|---|---|---|---|---|---|
| **E** | **N** | **C** | **R** | **Y** | **P** |
| **T** | **I** | **O** | A | B | D |
| F | G | H | J | K | L |
| M | Q | S | U | V | W |
| X | Z | 0 | 1 | 2 | 3 |
| 4 | 5 | 6 | 7 | 8 | 9 |

**Figure 3**

**Keyword: DIAMONDS**

| | | | | | |
|---|---|---|---|---|---|
| **D** | **I** | **A** | **M** | **O** | **N** |
| **S** | B | C | E | F | G |
| H | J | K | L | P | Q |
| R | T | U | V | W | X |
| Y | Z | 0 | 1 | 2 | 3 |
| 4 | 5 | 6 | 7 | 8 | 9 |

**Figure 4**

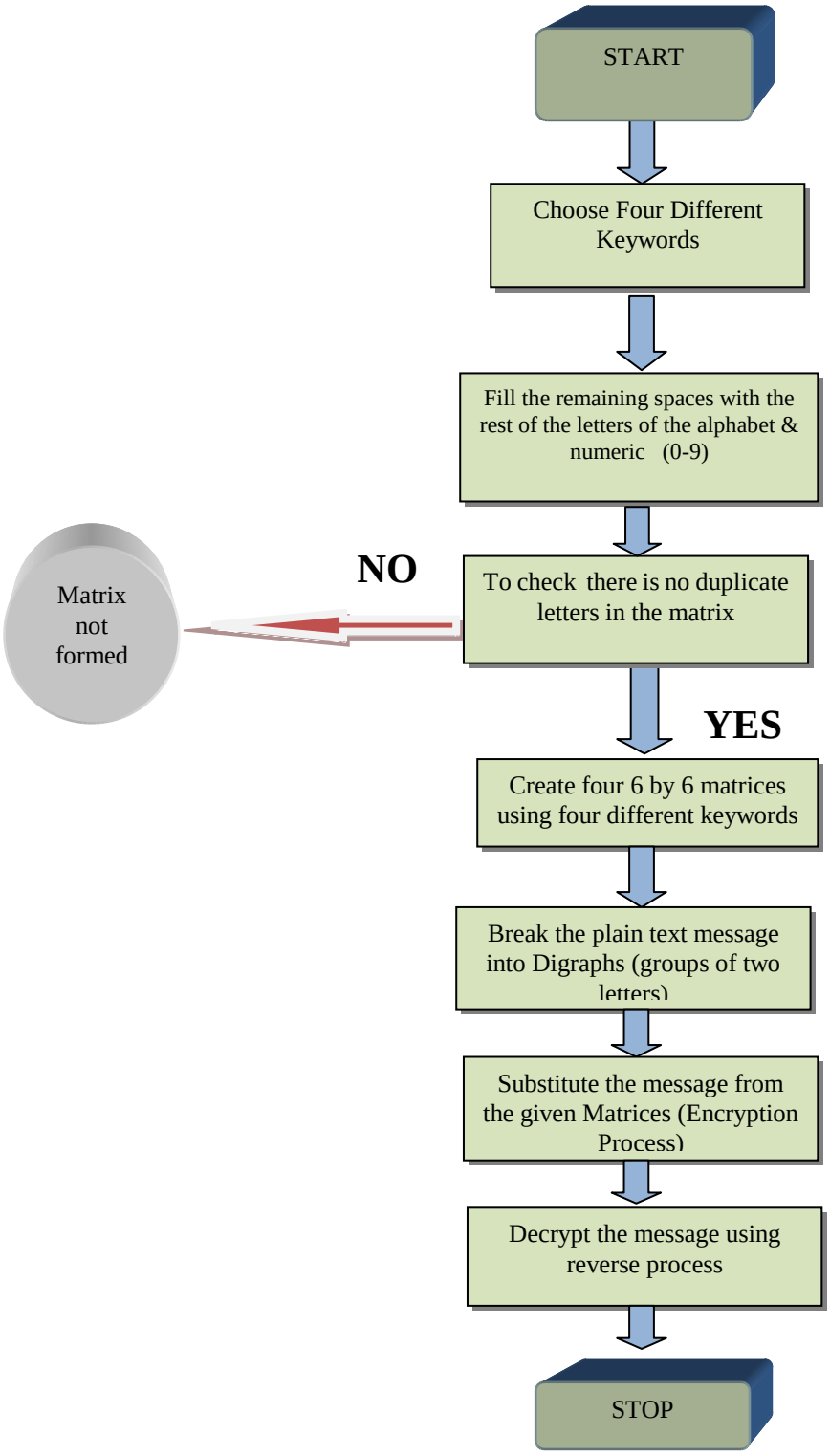
### *B. Algorithm*

- First we take input message which is user defined.
- If any space or punctuations occurs, then it should be automatically removed from the input message.
- After that we check any double occurrence, and then add “X” automatically in between these two characters.
- After removing the unwanted space and punctuations we get a modified message that is called the digraph message.
- Next we encrypt this digraph message with the keyword “MONARCHY”.
- After that corresponding three iteration steps introduced with three different keywords: “PLAYFAIR”, “ENCRYPTION” and “DIAMONDS”.
- During encryption process if any two character occurs same row or same column and any one of the character occurs at the last column(for same row character) or at the last row(for same column character) then in the encrypted message they becomes first column character(for same row character) or first row character(for same column character).
- Next we decrypt the last encrypted message with keyword “DIAMONDS” and repeat the same decryption process three times with three different keywords: “ENCRYPTION”, “PLAYFAIR” and “MONARCHY” respectively.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 1, January 2014

- During decryption process if any two character occurs same row or same column and any one of the character occurs at the first column(for same row character) or at the first row(for same column character) then in the encrypted message they becomes last column character(for same row character) or last row character(for same column character).
- Finally we recover the original text message which we give at the outset.

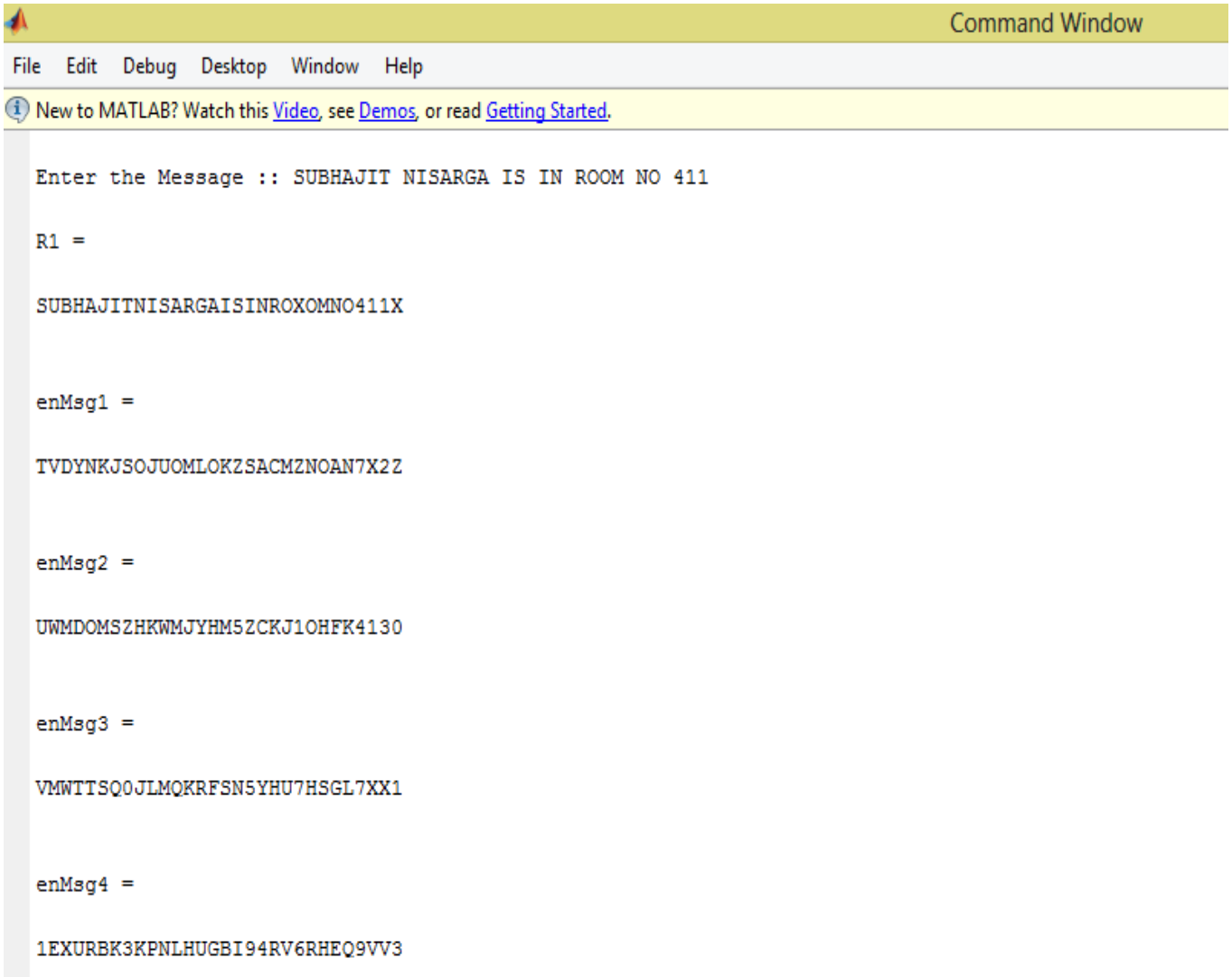
### *C. Flowchart*



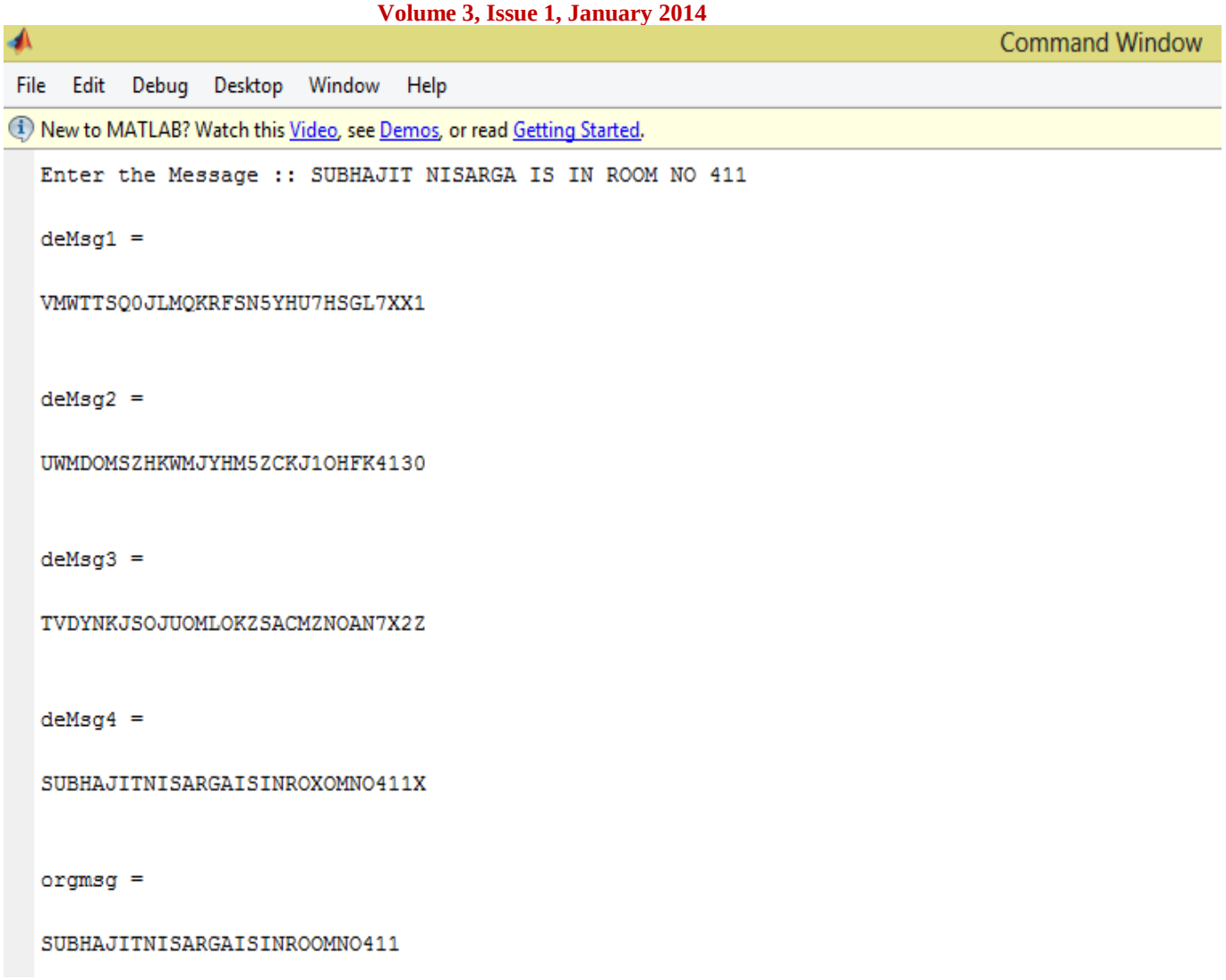
ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 1, January 2014

## IV. EXPERIMENTAL RESULTS

In this Paper for implementation of techniques MATLAB 7.0.2 version is used. MATLAB® is a high-performance language for technical computing. In our experiment we have used four different keywords and with the help of this four keywords we have encrypt and decrypt the text messages successfully. Here we include two figures. The original text message with its encrypted four versions is shown clearly in the first figure while in the second figure the four corresponding decrypted messages with the last original recovered text message is shown.



ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 1, January 2014




## V. FUTURE WORK

This extended play fair algorithm is based on the use of four 6 X 6 matrices of letters constructed using corresponding four keywords. The matrix is constructed by filling in the letters of the keyword from left to right and from top to bottom, and the filling in the remainder of the matrix with the remaining letters in alphabetic order and digits in ascending order form 0 to 9. We can extend this concept by using 16 by 16 matrix which includes not only alphanumeric characters but also special characters. Finally, we can make this 16 by 16 matrix concept stronger by using several iteration steps.

## VI. CONCLUSION

In this paper, we have used the basic playfair cipher and developed an enhanced playfair cipher dropping the restricting of previous playfair cipher using 5x5 matrix. The result of cipher is looking hard than the previous ciphers. The encrypted as well as decrypted messages with four different keywords are shown in experimental results.

## REFERENCES

[1] I. F. Akyildiz et al., "A Survey on Sensor Networks," IEEE Communications Magazine, August 2002, pp. 102-114.

[2] X.-Y. Li, et al., "Coverage in Wireless Ad Hoc Sensor Networks," IEEE Trans. on Computers, vol. 52, pp. 753-763 (June 2003).

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 1, January 2014

[3] I.Branovic, R. Giorgi, and E. Martinelli. Memory Performance of public-Key cryptography Methods in Mobile Environments. In ACM SIGARCH Workshop on Memory performance: Dealing with Applications, systems and architecture (MEDEA-03), pages 24–31, New Orleans, LA, USA, and September 2003.

[4] http://en.wikipedia.org/wiki/Playfair_cipher.

[5] William Stallings, "Cryptography and Network Security: Principles and Practice", 4th Edition, Prentice Hall, 2006.

[6] Derek Bruff, Ph.D, The Playfair Cipher Revealed Wynne MLAS 280-07 Cryptography July 13, 2009.

[7] Dr. Bruff, Playfair Cipher. FYWS Cryptology October 27, 2010.

[8] V. Umakanta Sastry, N. Ravi Shankar, and S. Durga Bhavani, Modified Playfair Cipher Involving Interweaving and Iteration. International Journal of Computer Theory and Engineering, Vol. 1, No. 5, December, 2009. 1793-8201.

[9] Kallam Ravindra Babu, Dr. S.Udaya Kumar, Dr. A.Vinaya Babu, An improved Playfair Cipher Cryptographic Substitution Algorithm, IJARCS, Volume 2, No-1, January February-2011,pages:211to214.

## AUTHOR BIOGRAPHY

**Nisarga Chand** obtained the Bachelor in Technology degree in Electronics & Communication Engineering and Master in Technology degree in Communication Engineering under West Bengal University of Technology, Kolkata, India from 2009 and 2011 respectively. He is presently serving as an Assistant Professor in Electronics and Communication Engineering Department at Mallabhum Institute of Technology; Campus: Braja-Radhanagar, P.O: Gosaipur, P.S: Bishnupur, Dist: Bankura - 722122, West Bengal, India. His research interest includes Wireless Sensor Network, Digital Communication and network security.

**Subhajit Bhattacharyya** received the Bachelor in Technology degree in Electronics & Communication Engineering and Master in Technology degree in VLSI & Microelectronics from West Bengal University of Technology Kolkata, India and Jadavpur University Kolkata, India from 2007 and 2011 respectively. He is currently an Assistant Professor at Mallabhum Institute of Technology Bishnupur, India. His research interest includes image processing, computer vision and network security.