International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.8 (2015), pp.187-198
http://dx.doi.org/10.14257/ijmue.2015.10.8.19

A Hybrid Polybius-Playfair Music Cipher

Chandan Kumar[1], Sandip Dutta[2], Soubhik Chakraborty[3]

[1,2]Department of CSE, Birla Institute of Technology, Mesra, Ranchi- 835215, India
[3]Department of Applied Mathematics, Birla Institute of Technology Mesra, Ranchi- 835215, India
[1]chandankr@bitmesra.ac.in, [2]sandipdutta@bitmesra.ac.in,
[3]soubhikc@yahoo.co.in

Abstract

Music has a versatile dimensionality; it can be used to express feelings, emotions and can also be used as a communicable language. Music and its attributes have been used in cryptography and steganography from a long time. Musical symbols and notes are used as replacement/substitution cipher. Using music as a cipher or cover media not only enhances the security of the message but also reduces its chance to be detected as an encoded or ciphered message. This paper proposes a hybrid Polybius and Playfair cipher which encodes the message into sequence of musical notes. The Playfair key matrix is generated using the Blum-Blum Shub generator. The bigrams of plain text message is first encrypted using Playfair cipher then individual character of the encrypted message is re-encrypted using Polybius cipher. The Playfair cipher enhances the security of the encrypted message over the simple substitution technique. The Polybius cipher then reduces the character set by appropriate number of symbols (here musical notes) for replacement. The basic 5X5 structure of key matrix in Polybius and Playfair is extended to 10X10 to hold the 95 prinTable characters of ASCII character set.

Keywords: *Musical cryptography, Polybius cipher, Playfair cipher, encryption, decryption*

1. Introduction

From the age of Julies Caesar various techniques have been applied to ensure secure communication [1]. These techniques can be classified as cryptography and steganography. [1] Cryptographic techniques use a cipher/encryption algorithm and a key to scramble the message in such a way that only intended parties can get the message back using the deciphering/decryption algorithm and the key used to decipher [29-33]. Encryption allows the original plain text data to be converted into unintelligible encrypted form also known as cipher text [29-32]. The cryptographic algorithms can be broadly classified according to their working and the key used for encryption and decryption process. Cipher algorithms are classified as block ciphers and stream cipher according to their working principles. The cipher algorithms which encrypt fixed length blocks at a particular time are classified as block cipher algorithm while others which encrypt the stream of a message are classified as stream cipher. Stream ciphers generally encrypt a particular character at a time. In block ciphers, if the length of the message is not an exact multiple of block size some padding bits are used which are discarded after the decryption process to get the exact message. Depending on the nature of the key used in encryption and decryption process the cipher algorithms can also be classified as symmetric and asymmetric key algorithms [1,31]. If same key is used for both encryption and decryption the process, the process is known as symmetric key cryptography while

ISSN: 1975-0080 IJMUE

asymmetric key cryptography (also called public key cryptography) uses two different keys namely public key and private key to encrypt and decrypt the message respectively. The key used in symmetric key cryptography should be pre-agreed by the sender and the intended receiver while in public key cryptography the public key of the receiver is announced to the public for encryption purpose and the receiver calculates his own private key which remains private to him and is used in decryption process. The main aim of cryptographic algorithms is to provide the basic security features which are confidentiality, integrity, authentication and non-repudiation.

Steganography based algorithms use a cover media to hide the content of the intended message into the cover message [34-35]. Steganography generally conceals the existence of message, so the communication is less prone to be suspected as hidden communication [35]. The early day steganography examples include secret inks, Morse code, microdots, use of different type faces, writing messages on the shaved head of soldiers *etc.* [36]. Modern day steganography uses digital media as a cover file to hide the intended message. The intended message is also known as payload data and the cover media is known as stego-media [35-37]. Due to the large file size and ample amount of redundancy, images, audio, video and execuTable files are used as cover media [37]. The intended message is hidden in the cover file by modifying the bits of the cover media according to the bits of intended message. The techniques used for modifying bit are LSB (least significant bit substitution), parity bit modification, echo hiding, DCT (discrete cosine transform), wavelet transforms, *etc.* [35-37]. Both steganography and cryptography have advantages and disadvantages over one another. Cryptography aims to encrypt the message in such a way that the encrypted cipher text should not be decrypted without the decryption key and guessing and trying all the possible keys (also known as brute force attack) should not be feasible with time as a constraint. Steganographic algorithms aim to hide the message in such a manner that the locations for the bits of intended message cannot be guessed or the encoding scheme used is secure over different type of attacks. Cryptography and steganography are used together to solve the purpose of information security in today's world.

2. Musical Cryptography

Music and its attributes have been used for encrypting message for a long time [3-12,23-28,36,38]. A brief literature survey on Musical cryptography was done by Eric Sam's [3]. Eric Sam's in his article "Musical Cryptography" says that many of the cryptologists have been great musician and mathematicians [3]. A strong relation between music and mathematics has been found. The early use of musical symbols as a replacement for the plain text characters dates back to 15^{th} century. Musical scores have been used to hide messages inside it. Tractus varii medicinales [10] used five different pitches in five different ordering which produced 25 symbols, which was later used as an alphabetic cipher [3]. Systems similar to Tractus were developed by 16^{th} century which used 9 different pitches to produce 72 symbols. Wilkin's invented a musical cipher system which used to represent alphabets by the minnums on the five lines in the musical score. Athanius Kircher [6] a polymath introduced the idea of orchestra to encrypt messages. Kircher used four different notes of six different musical instruments yielding 24 musical notes. Leibniz [7-8,11] introduced the idea of a superficial language containing of notes and pitches. Hooper and Kluber [5] used rotating cipher wheels to encrypt messages. The cipher wheel of Hooper and Kluber had musical notes and their corresponding letters written on two concentric circles, the device permitted resetting so the ciphered message had different notation at different settings.

Schumann [23-24] has also used the idea derived from Kluber, which helped him devising a three lines and eight notes cipher system. Bach [9-10] and Elgar [25] used to write the names of their friends in musical style. Elgar [25] sent various messages to his friends which were written in musical notation. Olivier Messiaen [12] used his own system of signs to encrypt messages using music. Olivier Messiaen's System of Signs comprises of a musical alphabet, a simple grammar and a series of leitmotifs, where the three of the above were used to transliterate text into music. Bishop Godwin's [28] "Lunatic Language" uses a music cipher to convey messages [38]. Kumar [22] used music to encode and decode any encrypted message. Dutta *et al.* [13] proposed a scheme of encrypting messages using 36 different frequencies; twelve notes each from three octaves. Dutta *et al.* [13] were able to encrypt the 26 characters of English alphab*et al*ong with 10 roman numerals. Dutta *et al.* [14] in their work used raga malkhauns for encrypting messages. The scheme used by Dutta *et al.* [14] used the transition probability of musical notes in encrypting messages. Dutta *et al.* [15] used mathematically generated notes for encrypting messages as a bigram. Dutta *et al.* in their work used the transition matrix for the bigram [15]. Dutta *et al.* [21, 39] in their work have used candidate notes for each character and used those candidate notes to find the best plausible musical sequence for the candidate notes of the plain text message as cipher message. Yamuna *et al.* [16] have used graph theory to encrypt messages using musical notes, they have also used the cipher feedback chaining for encryption purpose. Maity [20] used magic squares for the permutation of characters for the Polybius square and used musical notes to encrypt messages. Glatfelter *et al.* [17] have proposed a framework for encrypting messages using frequencies of musical notes. Glatfelter *et al.* [17] have also used the matrix multiplication as a base for obtaining the cipher message. Lee *et al.* [18] have proposed a rhythm key based encryption scheme for Ubiquitous Devices, where they have used musical rhythms as the cryptographic key. Yamuna [19] has encrypted messages using musical notes.

3. Hybrid Polybius-Playfair Cipher

3.1 Polybius Cipher

Polybius cipher is a substitution cipher devised by an ancient Greek historian Polybius. [29-32] The Polybius cipher fractionates the character set to represent it with smaller number of symbols. The letters of the alphabet are arranged in a 5x5 Polybius square, where the letters are identified by their grid position *i.e.* by the row and column position or simply the coordinates. Encryption is simply done by replacing plain text characters with corresponding pair of numbers. To represent the English alphabet using Polybius square the characters I and J share the same location which can be easily identified at the time of decryption by the meaning of the text. The Polybius cipher can be keyed also, where the letters of key are inserted first into the square without repetition and the remaining letters are inserted sequentially. Polybius cipher can take form of polyalphabetic substitution cipher by taking a long key and encrypting the key with the square and taking the sum with the encrypted plain text message.

Plaintext: This is a secret message
Ciphertext: 44232443 2443 11 431513421544 32154343112215

Table 1. Key Matrix for Polybius Cipher

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I/J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

3.2 Playfair Cipher

Playfair is a polyalphabetic substitution cipher invented by Wheatstone and promoted by Playfair [30-32]. The cipher is based on a 5X5 square which can accommodate 25 letters. As there is space for only 25 letters in the square so the letter J is dropped and is replaced by the character I or II. The letters in the square are arranged by taking a key and inserting the letters of the key without repetition and then the remaining letters are appended. The encryption of message is done by encrypting a pair of letters at a time. The whole message is divided into pairs of letters, if the length of the message is not even then a filler 'x' is inserted. If both of the letters of pair are same then a filler x is inserted and to compensate this an extra x is inserted at the end of the message. To encrypt the plain text message, the message is broken into digraphs (groups of 2 letters) and mapped out using the Playfair key Table. The mapping rules are:

1. If the letters of the digraph appear in the same row of the key Table, replace them with letters to their immediate right respectively (if the letter of the original pair is rightmost element in the row, wrap around to left of the row).
2. If the letters of the digraph appear in the same column of the key Table, replace them with the letters immediately below respectively (if a letter in the original pair is on the bottom of the column, wrap around to the top side the column).
3. If the letters of the digraph are not on the same row or column, replace them with the letters on the same row of the letter and corresponding column of the other letter of the pair. The order is important thus the first letter of the encrypted pair is the one that lies on the same row as the first letter and the column of the second letter of the plaintext pair.

Table 2. Key Matrix for Playfair Cipher

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

Plain Text Message: HELLO WORLD
Playfair message: HE LX LO WO RL DX
Playfair Cipher: KG YV RV VQ GR ZC

The decryption rules are same as the encryption. The cipher message is mapped with the same Playfair matrix to get the plain text message.

3.3 Hybrid Polybius Playfair Cipher

At first the digraph is encrypted using the Playfair cipher, the encrypted cipher message is then re-encrypted using the Polybius cipher. The decryption process is the reverse of encryption, where the cipher message is decrypted first by Polybius cipher then the decrypted message is again decrypted using Playfair cipher.

Cipher text = $E_{Polybius}$ ($E_{Playfair}$ (Plaintext))

Plain text = $D_{Playfair}$ ($D_{Polybius}$ (Ciphertext))

Plain Text Message:	HELLO WORLD					
Playfair message:	HE	LX	LO	WO	RL	DX
Playfair Cipher:	KG	YV	RV	VQ	GR	ZC
Polybius Cipher:	CDCB	ADEB	BBEB	EBDC	CBBB	EEBD

Table 3. Key Matrix for Modified Polbius-Playfair Cipher

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

The key matrix in the proposed scheme is a 10x10 matrix (refer Table 8) where the corresponding numerical equivalent of the letters is shown in the Table 4. To fill the key matrix completely we need to insert five extra characters, so we have chosen to insert five extra spaces whose numerical equivalents are 96 to 100. These numeric equivalents are jumbled using the proposed permutation scheme and then inserted into the key matrix.

3.4 Blum Blum Sub Generator

Random number generation is a bit tricky task in any cryptographic algorithm. Random number generator functions, basically pseudo random generators are based on recurrence relationships, linear congruential generators and one way trapdoor functions *etc*. Blum Blum Sub Generator [40] is used to generate pseudo random sequence of random numbers and random bits. This pseudo random number generator starts with taking two prime numbers p and q whose mod with 4 is equal to 3. We find the product of the two number and call it M. Then a number co-prime to M is taken and called x_0, where x_0 is a quadratic residue modulo M. The Blum Blum Generator is defined as $b_i=S_i \bmod 2$. The starting value of the sequence is calculated as:

$S_0 = (x_0)^2 \bmod(M)$.

$b_0=S_0 \bmod(2)$,

The rest of the sequence is found using

for i= 1 to n

$S_i = (S_{i-1})^2 \bmod (M)$.

$b_i=S_i \bmod(2)$

end

The random number generator generates random numbers S_i in the field M, a care should be taken while choosing p, q and M so that the number of terms generated in the set S is large enough. The numbers generated as S_i repeats the same pattern.

This random number generator can be used for permutation purpose also. If we get *S* to be a large set and it consists of all the numbers between 1 and p. then the sequence generated by S mod (p) is random permutation of numbers from 1 and p. If the set S has length less than the permutation set then we find all the elements of permutation set which are not present in S and call it the set "Left". We find the length of the set left and start inserting the elements of the "Left" set in S by initializing pos=length(left+7) and resetting the position for each iteration by pos=mod(pos+i^2 ,length(left))+1 and $S_{length+i}$= Left(pos) and delete the element from the Left set. And reiterate it to length times.

Table 4. Numeric Equivalents for the PrinTable Characters Used in Encryption

1	'space'	21	4	41	H	61	\	81	p
2	!	22	5	42	I	62	]	82	q
3	"	23	6	43	J	63	^	83	r
4	#	24	7	44	K	64	'under score'	84	s
5	$	25	8	45	L	65	`	85	t
6	%	26	9	46	M	66	a	86	u
7	&	27	:	47	N	67	b	87	v
8	'	28	;	48	O	68	c	88	w
9	(	29	<	49	P	69	d	89	x
10	)	30	=	50	Q	70	e	90	y
11	*	31	>	51	R	71	f	91	z
12	+	32	?	52	S	72	g	92	{
13	,	33	@	53	T	73	h	93	|
14	-	34	A	54	U	74	i	94	}
15	.	35	B	55	V	75	j	95	~
16	/	36	C	56	W	76	k	96-100	'space'
17	0	37	D	57	X	77	l		
18	1	38	E	58	Y	78	m		
19	2	39	F	59	Z	79	n		
20	3	40	G	60	[	80	o		

3.5 Proposed Permutation Algorithm

Choose two prime numbers p,q such that p(mod 4) = q(mod 4) = 3 and p!=q. choose a seed value x_0 such that x_0 is co-prime to pxq call it M. select the permutation range R with Set R having numbers 1 to R.

Step1: Initialize *p*, *q*, x_0, *M*. Select permutation range *R*.

Step2: Now calculate $S_0 = (x_0)^2 \bmod(M)$

Step3: For I = 1 to 10000

$$S_i = (S_{i-1})^2 \bmod(M)$$

End

Step4: Find unique ($S_i$mod(R))s without changing the order of occurrence and call it S

Step5: Find LeftR = (Set R- S) (Numbers in R which are not in S)

Step6: Permute LeftR as

Initialize pos=length(left+7);

For i=1 to length(LeftR)

pos=mod(*pos*+i^2 ,*length(left)*)+1;

$LeftR_2$(i)= LeftR(pos);

Delete LeftR(mod(pos+i^2 ,length(left)));

Recalculate length(left); (length reduces as each iteration inserts one element of the set LeftR in Left R_2 and the element is deleted form the set LeftR)

End

Step7: Concatenate Unique S with Left R_2.

Example 1: Choosing the value of p and q as 107 and 839 respectively and setting the seed x_0 =11 gives the sequence of random number which when taken a mod of 100 produces the list(refer Table 5). As the random number generated are in the range 0-99 adding it with one gives us the desired permutation set.

Table 5. Permutation of Numbers Through 0 to 99 for Example 1

p= 107,			**q=839,**			**x0 = 11**										
21	41	30	2	1	44	72	63	34	59	18	69	96	45	6	40	76
52	36	17	20	93	9	35	10	32	13	80	27	79	54	92	3	0
67	60	91	95	48	28	39	89	74	98	78	43	57	90	42	25	53
29	47	82	73	16	88	4	51	61	23	22	65	62	33	24	19	26
55	68	77	15	87	56	66	94	31	58	7	38	14	49	86	70	97
46	75	71	85	50	8	84	83	5	37	81	99	11	64	12		
No number is left between 0-99																

Example 2: Choosing the value of p and q as 67 and 811 respectively and setting the seed x_0 =439 gives the sequence of random numbers which when taken a mod of 100 produces the list randoms (refer Table 6). The remaining numbers are found as a set Left. The elements of the Left set are permutated according to the proposed scheme and concatenated to the list randoms.

First we have found the unique random numbers generatd using Blum Blum generator and then we dive it with 100 and found the modular remainder of the numbers in a unique sequence. The number of the Range R which were not present in the unique sequence were found and were added to the sequence using the proposed permutation algorithm.

Table 6. Permutation of Numbers Through 0 to 99 for Example 2

p= 67,			***q=811,***			***x_0= 439***										
Unique Random mod (100) = randoms																
10	72	66	11	54	23	15	31	94	38	0	8	9	85	26	86	28
50	16	75	27	5	92	44	46	79	53	18	49	30	22	91	3	80
36	69	25	61	62	52	68	24	71	41	42	48	40	73	88	82	89
19	84	59	47	95	1	60	17	2	78	7	65	96	58	90	97	
Left =																
4	6	12	13	14	20	21	29	32	33	34	35	37	39	43	45	51
55	56	57	63	64	67	70	74	76	77	81	83	87	93	98	99	
Permutation (0-99) =																
10	72	66	11	54	23	15	31	94	38	0	8	9	85	26	86	28
50	16	75	27	5	92	44	46	79	53	18	49	30	22	91	3	80
36	69	25	61	62	52	68	24	71	41	42	48	40	73	88	82	89
19	84	59	47	95	1	60	17	2	78	7	65	96	58	90	97	6
29	56	14	4	39	33	76	12	35	64	32	43	21	13	55	63	83

98	74	77	34	51	20	81	57	87	70	37	67	45	99	93

Example 3: Using the Playfair key "Playfair matrix" we first insert the letters of the Playfair key first without repetition. Then we insert the remaining permuted sequence without repetition. Here we choose the value of p and q as 319 and 83 respectively and set the seed x_0 =7. The sequence of random numbers which when taken a mod of 100 produces the list randoms (refer Table 7). The remaining numbers are found as a set Left. The elements of the Left set are permutated according to the proposed scheme and concatenated to the list randoms. The random permutation sequence is appended to the letters of the key matrix without repetition.

Playfair key: "Playfair matrix"
P=49, l=77, a=66, y=90, f=71, a=66, i=74, r=83, ' '=100, m=78, a=66, t=85, r=83, i=74, x=89
matrix = 49 77 66 90 71 74 83 100 78 85 89

Table 7. Permutation of Numbers Through 1 to 100 for Example 3

p = 319, q =83, x_0 = 7																
Unique randoms mod (100) =																
49	1	92	52	0	5	45	90	78	74	23	8	62	3	55	56	16
68	75	17	87	79	53	13	66	6	39	35	24	69	81	94	29	36
95	48	40	31	44	67	96	85	47	84	42	7	97				
left =																
2	4	9	10	11	12	14	15	18	19	20	21	22	25	26	27	28
30	32	33	34	37	38	41	43	46	50	51	54	57	58	59	60	61
63	64	65	70	71	72	73	76	77	80	82	83	86	88	89	91	93
98	99															

Permutation(1-100) = adding (randoms+1) in the sequence and then appending permuted set (left+1) in the sequence																
50	2	93	53	1	6	46	91	79	75	24	9	63	4	56	57	17
69	76	18	88	80	54	14	67	7	40	36	25	70	82	95	30	37
96	49	41	32	45	68	97	86	48	85	43	8	98	5	16	33	66
23	100	12	52	34	64	51	92	15	10	83	62	47	58	87	72	19
28	38	65	26	42	55	74	35	94	81	39	29	21	78	89	59	84
27	73	13	22	60	20	44	3	90	11	71	99	61	31	77		
Inserting the elements of the Playfair base key in the matrix and then adding the rest of the elements sequentially, the sequence of elements of Key Matrix in row major order =																
49	77	66	90	71	74	83	100	78	85	89	50	2	93	53	1	6
46	91	79	75	24	9	63	4	56	57	17	69	76	18	88	80	54
14	67	7	40	36	25	70	82	95	30	37	96	41	32	45	68	97
86	48	43	8	98	5	16	33	23	12	52	34	64	51	92	15	10
62	47	58	87	72	19	28	38	65	26	42	55	35	94	81	39	29
21	59	84	27	73	13	22	60	20	44	3	11	99	61	31		

Notes as Polybius Indices=['S', 'r', 'R', 'g', 'G', 'M', 'P', 'd', 'D', 'N']; these indices can be chosen from any of the 12 chromatic notes of the western or Indian music or there equivalent system.

Table 8. Key Matrix from Permutation Table of Example 3

	'S'	***'r'***	***'R'***	***'g'***	***'G'***	***'M'***	***'P'***	***'d'***	***'D'***	***'N'***
'S'	P	l	a	y	f	i	r	space	m	t
'r'	x	Q	!	|	T	space	%	M	z	n
'R'	j	7	(	^	#	W	X	0	d	k
'g'	1	w	o	U	-	b	&	G	C	8
'G'	e	q	~	=	D	space	H	?	L	c
'M'	space	u	O	J	'	space	$	/	@	6
'P'	+	S	A	_	R	{	.	)	]	N
'd'	Y	v	g	2	;	E	`	9	I	V
'D'	B	}	p	F	<	4	Z	s	:	h
'N'	,	5	[	3	K	"	*	space	\	>

3.6 Encryption Process

1. Generate the key matrix using the seed values for Blum Blum Generator and optional keys (Playfair key, Polybius Indices Key or rhythm key).
2. Convert the message into Playfair digraphs and encrypt the digraphs using Playfair algorithm and the generated key matrix.
3. Encrypt the intermediate Playfair cipher using the Polybius indices of the same key matrix.
4. Generate the music file using the generated musical equivalents of the Polybius cipher.

The decryption process will be same as of encryption but in the reverse order. The music file will be read first and the musical notes will be taken into pairs and converted to the Polybius message. The intermediate Polybius message is then converted back to Playfair message using the Playfair algorithm and the key matrix. The key matrix will be generated by the same process as it was done in encryption side.

Plain Text Message:

Music has a versatile dimensionality; it can be used to express feelings, emotions and can also be used as a communicable language.

Playfair Cipher:

Q/4‘space’eHpt4/ymYq‘space’ZyPraqHWmPLMhab!tarPfE'rPHet!W‘space’qH/}L jmPb!‘space’jZa?B4/PDqPt‘space’9p5\LP8aabMhmyzkHet!my‘space’}b!1‘space’$O B?Wz‘space’pmyHeCal@‘space’t~twiqHay!VOlY~]%

Polybius Cipher:

rrMdDMSdGSGPDRSNDMMdSgSDdSGrSdDPSgSSSPSRGrGPRMSDSSGDrd DNSRgMrRSNSRSPSSSGdMMGSPSSGPGSSNrRRMGMGrGPMdDrGDRSSDSS gMrRMSRSDPSRGdDSDMMdSSGGGrSSSNrMddDRNrNDGDSSgNSRSRgMrdD NSDSgrDRNGPGSSNrRSDSgSdDrgMrRgSGMMPMRDSGdRMrDSdDRSDSgGP GSgDSRSrMDrMSNGRSNgrSMGrGPSRSgrRdNMRSrdSGRPDrP

4. Key Matrix and the Security Issues

The Playfair matrix for the proposed scheme is constructed by inserting the letters of the Playfair key first without repetition; the remaining letters are inserted using the sequence generated by the proposed Blum-Blum Shub generator technique for permutation. The letters are inserted by filling the columns of the first row first then the corresponding rows in order. If no key has been used for the Playfair matrix, the Blum Blum Shub generator scheme for permutation generates a random permutation of letters for the Playfair matrix. The indices for the Polybius cipher can be either set as default or can also be keyed. The key used for the Polybius matrix is a rhythm key, where the indices are used without repetition. For example fixed indices can be "C Db D Eb E F F# G Ab A Bb B" and a rhythm key can be ABaAabAB where the indices are ABab without repeating any note.

The number of different sequences generated by the Blum-Blum shub generator cannot achieve the value near 100!, but the use of Playfair key helps it to get it near to that value which makes it cumbersome to try all the possible keys to decrypt the encrypted message.

The proposed scheme also helps in preserving the spaces in the plain text message. The spaces in the plain text message can be encrypted in different ways also; we can use any of the 6 permitted values for the spaces. But a care should be taken in implementing the algorithm, because while decrypting the digraph the decryption algorithm could not identify which one of the all 6 candidate space is used in decrypting the digraph. An alternate implementation can solve this problem by using the numerical equivalents of the characters; the six candidate spaces will have six different numerical equivalents.

The rhythm key also increases the difficulty in guessing the key as it can take any one of the possible 10! ways. The proposed scheme can also be implemented to permit the user a choice for using or not the Playfair key and the Rhythm key.

5. Implementation, Results and Discussion

The proposed algorithm is implemented using Matlab®. The implementation provides the user a choice to either provide or not the Playfair key and the Rhythm key. The values of p, q and x_0 are necessary for the Blum-Blum Shub generator for permuting the key matrix. The user has to remember the values p,q,x_0, the rhythm key and the Playfair key. These values are used on both sender and receiver side to generate the key matrix for encryption and decryption process. The produced musical sequence is written as a MIDI file. The subsequent MIDI file is sent as the encrypted message to the receiver. The receiver decrypts the MIDI file by reading the notes of the MIDI and applying the reverse of the encryption process.

The encrypted message produces a satisfactory musical sequence in the form of a midi sequence. The proposed system lacks in aesthetic appeal, evolutionary algorithm can be used to make the musical sequence sound better by improvising the tempo and the rhythm. Musical cryptography can be used to generate motifs which are in turn a musical cryptogram. Musical cryptography can also be used as a replacement of audio steganography by reducing the effort of finding appropriate sized files as a cover message. The bandwidth under use due to the cover file can also be reduced by generating musical notes of desired length. Cryptography using music can further reduce the chance of cipher message to be detected as cipher. Some better algorithms which can produce real world music sequences as a cipher message is demanded as a future scope.

References

[1] D. Kahn, "The Codebreakers: The Comprehensive History Of Secret Communication From Ancient Times To The Internet," **(1996)**.
[2] D. Davies, "A brief history of cryptography," Information Security Technical Report, vol. 2 no. 2, **(1997)**, pp.14-17.
[3] E. Sams, "Musical cryptography," Cryptologia, vol. 3 no. 4, **(1979)**, pp.193-201.
[4] E. Sams, "Elgar's Cipher Letter to Dorabella," The Musical Times, vol. 111 no. 1524, **(1970)**, pp. 151-154.
[5] J. L. Klüber, "Kryptographik," **(1809)**.
[6] A. Kircher, "Musurgia universalis," 1650, **(1988)**.
[7] A. P. Coudert, R. H. Popkin and G. M. Weiner, "Eds. Leibniz, mysticism and religion," International Archives of the History of Ideas, Springer, vol. 158, **(1998)**.
[8] J. B. Kennedy, "The concise Oxford dictionary of music," OUP Oxford, **(2004)**.
[9] R. Tatlow, "Bach and the Riddle of the Number Alphabet". Cambridge University Press, **(1991)**.
[10] S. E. Sadie, "The new Grove dictionary of music and musicians," **(1980)**.
[11] J. Daverio, "Crossing Paths: Schubert, Schumann, and Brahms," Oxford University Press, **(2002)**.
[12] A. Shenton, "Olivier Messiaen's system of signs: notes towards understanding his music," Ashgate Publishing, Ltd., **(2008)**.
[13] S. Dutta, S. Chakraborty and N. C. Mahanti, "A novel Method of Hiding Message Using Musical Notes," International Journal of Computer Application, vol.1 no.16, **(2010)**.
[14] S. Dutta, S. Chakraborty and N. C. Mahanti, "Using Raga as a Cryptographic Tool," Advances in Network Security and Applications, Communications in Computer and Information Science, CNSA 2011 (Springer), vol. 196, **(2011)**.
[15] S. Dutta, C. Kumar and S. Chakraborty, "A Symmetric Key Algorithm for Cryptography using Music," International Journal of Engineering and Technology, vol. 5 no. 3, **(2013)**.
[16] M. Yamuna, A. Sankar, S. Ravichandran and V. Harish, "Encryption of a Binary String Using Music Notes and Graph theory," International Journal of Engineering and Technology, vol. 5 no. 3, **(2013)**.
[17] J. W. Glatfelter and C. W. Raab, "Cryptography using a symmetric frequency-based encryption algorithm," U.S. Patent No. 8,855,303. 7, Oct. **(2014)**.
[18] J. D. Lee, H. J. Im, W. M. Kang, & J. H. Park, "Ubi-RKE: A Rhythm Key Based Encryption Scheme for Ubiquitous Devices", Mathematical Problems in Engineering, **(2014)**.
[19] Y. Manimuthu, "Insertion Method Using Music Notes," Innovare Journal of Engineering & Technology, **(2014)**.
[20] M. Maity, "A Modified Version of Polybius Cipher Using Magic Square and Western Music Notes," International Journal for Technological Research in Engineering, vol. 1 no.10, **(2014)**.
[21] C. Kumar, S. Dutta, and S. Chakraborty, "Hiding Messages using Musical Notes: A Fuzzy Logic Approach," International Journal of Security and Its Applications, vol. 9 no. 1, **(2015)**, pp. 237-248.
[22] V. R. Kumar, "Coding Encrypted Messages into Music," Diss. Carleton University, **(2006)**.
[23] E. Sams, "Did Schumann Use Ciphers? " The Musical Times, **(1965)**, pp.584-591.
[24] E. Sams, "The Schumann Ciphers," The Musical Times, **(1966)**, pp. 392-400.
[25] E. Sams, "Elgar's Enigmas," Music & Letters, **(1997)**, pp. 410-415.
[26] K. B. C. Uhde, "Psychologische Musik, Joseph Joachim, and the Search for a New Music Aesthetic in the 1850s." **(2014)**.
[27] T. Allen, D. Branscombe and J. Bury, "Ciphers and Commuting Algebras of Hilbert Spaces in Music," **(2012)**.
[28] H. N. Davies, "Bishop Godwin's' Lunatique Language," Journal of the Warburg and Courtauld Institutes, **(1967)**, pp. 296-316.
[29] D. Salomon, "Data Privacy and Security: Encryption and Information Hiding," Springer, **(2003)**.
[30] C. Christensen, "Review of Secret History: The Story of Cryptology by Craig P. Bauer," Cryptologia, vol. 38 no. 2, **(2014)**, pp. 192-193.
[31] D. Salomon, "Elements of Cryptography," Foundations of Computer Security. Springer London, **(2006**), pp. 263-284.
[32] D. Salomon, "Coding for data and computer communications," Springer, **(2006)**.
[33] D. Kahn, "The codebreakers," Weidenfeld and Nicolson, **(1974)**.
[34] S. Katzenbeisser and F. Petitolas, "Information hiding techniques for steganography and digital watermaking," DNA, vol. 28 no. 2, **(2004)**.
[35] N. Provos and P. Honeyman, "Hide and seek: An introduction to steganography," Security & Privacy, IEEE, vol. 1 no. 3, **(2003)**, pp. 32-44.
[36] T. Leary, "Cryptology in the 15th and 16th Century," Cryptologia, vol. 20 no. 3, **(1996)**, pp. 223-242.
[37] D. Artz, "Digital steganography: hiding data within data," Internet computing, IEEE vol. 5 no. 3 **(2001)**, pp. 75-80.
[38] M. B. Campbell, "Speedy Messengers: Fiction, Cryptography, Space Travel, and Francis Godwin's The Man in the Moone," Yearbook of English Studies, vol. 41 no.1, **(2011)**, pp. 190-204.

[39] C. Kumar, S. Dutta and S. Chakraborty, "Musical Cryptography using Genetic Algorithm," Circuit, Power and Computing Technologies (ICCPCT), 2014 International Conference on. IEEE, **(2014)**, pp. 1742-1747.
[40] C. Ding, "Blum-blum-shub generator," Electronics Letters, vol. 33 no. 8, **(1997)**, pp. 677-677.

Authors

Chandan Kumar, is research scholar in the Department of Computer Science and Engineering Birla Institute of Technology, Mesra, Ranchi. His areas of interest are Cryptography, Network Security and Biometrics.

Sandip Dutta, a PhD in Computer Science, is Head of Department Computer Science and Engineering, BIT Mesra, Ranchi, India. His areas of interest are Cryptography, Network Security, Biometrics and Software Engineering. He has been guiding PhD scholars in the areas of cryptography and Software engineering.

Soubhik Chakraborty, a PhD in Statistics, is an associate professor in the department of Applied Mathematics, BIT Mesra, Ranchi, India. He has published several papers in algorithm and music analysis and is guiding research scholars in both the areas. He is a reviewer of prestigious journals like Mathematical Reviews (American Mathematical Society), Computing Reviews (ACM) and IEEE Transactions on Computers *etc* besides being the Principal Investigator of a UGC major research project on music analysis in his department he is also an amateur harmonium player.

