CHAPTER 3

A MODIFIED HILL CIPHER FOR A LARGE BLOCK OF PLAINTEXT WITH INTERLACING AND ITERATION
3.1. INTRODUCTION

In the previous chapter, we have developed a block cipher using a large key matrix. In this, we have used interlacing of the binary bits of the plaintext vectors, occurring in the plaintext matrix, as the primary concept. Here, the multiplication of the plaintext with the key causes diffusion and the interlacing of the plaintext at various stages of iteration causes confusion in an effective manner.

In the present chapter, our objective is to develop a block cipher, wherein the block is taken in the form of a large matrix. In this, we illustrate the cipher by giving a pair of examples. In the first one, the plaintext block is taken in the form of an 8×8 matrix and in the second one, it is taken as a whole in the form of a matrix which has 8 rows and any number of columns, depending on the size of the entire plaintext.

3.2. DEVELOPMENT OF THE CIPHER

Consider a plaintext. Let us use ASCII code and represent it in the form of a matrix of size nxm. Thus we have

\[ P = [P_{ij}] , \text{ } i = 1 \text{ to } n, \text{ } j = 1 \text{ to } m \]  

(3.2.1)

Let the key matrix K be given by

\[ K = [K_{ij}] , \text{ } i = 1 \text{ to } n, \text{ } j = 1 \text{ to } n. \]  

(3.2.2)

Following Hill, the process of encryption is described by using the relation

\[ C = KP \mod 128 \]  

(3.2.3)

The process of decryption is governed by the relation
\( P = K^{-1}C \mod 128, \quad (3.2.4) \)

where \( k^{-1} \) is the modular arithmetic inverse of \( K \).

In the present analysis, we include interlacing (decomposition) in the process of encryption (decryption) and use iteration in both encryption and decryption. Here, it is to be noted that decomposition is a reverse process to that of interlacing.

Let us now illustrate the process of interlacing. For simplicity, consider a plaintext matrix of size 8\times8 given by

\[
P = [p_{ij}], \quad i = 1 \text{ to } 8, \quad j = 1 \text{ to } 8 \quad (3.2.5)
\]

Writing each number of (3.2.5) in its binary form, we get \([P_{ij}] = [b_{il}]\)

Where, \( i = 1 \text{ to } 8, \ \ l = 1 \text{ to } 7, \ \ j = 1 \text{ to } 8.\)

Typically we can write the first column of the matrix \([P_{ij}]\) as follows:

\[
\begin{pmatrix}
    b_{12}^1 & b_{13}^1 & b_{14}^1 & b_{15}^1 & b_{16}^1 & b_{17}^1 \\
    b_{22}^1 & b_{23}^1 & b_{24}^1 & b_{25}^1 & b_{26}^1 & b_{27}^1 \\
    b_{32}^1 & b_{33}^1 & b_{34}^1 & b_{35}^1 & b_{36}^1 & b_{37}^1 \\
    b_{42}^1 & b_{43}^1 & b_{44}^1 & b_{45}^1 & b_{46}^1 & b_{47}^1 \\
    b_{52}^1 & b_{53}^1 & b_{54}^1 & b_{55}^1 & b_{56}^1 & b_{57}^1 \\
    b_{62}^1 & b_{63}^1 & b_{64}^1 & b_{65}^1 & b_{66}^1 & b_{67}^1 \\
    b_{72}^1 & b_{73}^1 & b_{74}^1 & b_{75}^1 & b_{76}^1 & b_{77}^1 \\
    b_{82}^1 & b_{83}^1 & b_{84}^1 & b_{85}^1 & b_{86}^1 & b_{87}^1
\end{pmatrix}
\]

Similarly, we can have the second column with superscript 2 on all the elements (instead of 1). In the same manner, we can write all the other columns.

Now, let us place the eight columns of \([P_{ij}]\) one after the other. Thus we get a matrix of size 8\times56, containing the elements of \([b_{ii}]\). Here, the process of interlacing can be described as follows. Let us focus our
attention on the fifty six elements of the first row of the matrix formed above. This set of elements is divided into two equal halves. The first bit of the second half is placed after the first bit of the first half, the second bit of the second half is placed after the second bit of the first half and so on. After mixing in this manner, we place these elements in the form of a matrix which is given below.

\[
\begin{pmatrix}
    b_{11}^1 & b_{12}^1 & b_{13}^1 & b_{14}^1 \\
    b_{15}^1 & b_{16}^1 & b_{17}^1 & b_{18}^1 \\
    b_{11}^2 & b_{12}^2 & b_{13}^2 & b_{14}^2 \\
    b_{15}^2 & b_{16}^2 & b_{17}^2 & b_{18}^2 \\
    b_{11}^3 & b_{12}^3 & b_{13}^3 & b_{14}^3 \\
    b_{15}^3 & b_{16}^3 & b_{17}^3 & b_{18}^3 \\
    b_{11}^4 & b_{12}^4 & b_{13}^4 & b_{14}^4 \\
    b_{15}^4 & b_{16}^4 & b_{17}^4 & b_{18}^4 \\
\end{pmatrix}
\]

Similarly we get seven more matrices by using the rows two to eight of the matrix of the size 8\times56 mentioned above.

Thus we get all the eight matrices having binary bits in each row. Basing upon these binary bits, we find the corresponding decimal numbers and hence obtain an 8\times8 matrix, which is including the elements of all the columns. This can be considered as the new plaintext matrix (obtained after interlacing).

In a similar manner, it is possible to interlace the plaintext matrix, even when we are having more number of columns.

The procedures interlacing and decomposition are used in encryption and decryption respectively. The development of the cipher is shown in the schematic diagram given in Fig. 3.1
In this analysis, N denotes the number of iterations and it is taken as 16.

Fig. 3.1.
3.3. ALGORITHMS

The algorithms required for encryption and decryption are designed as follows:

3.3.1 ALGORITHM FOR ENCRYPTION

\{ 
1. Read n,N,K,P;
2. P0 = P;
3. for i = 1 to N 
   \{ 
   Pi = KP^{i-1} \text{ mod } 128;
   \text{Interlace}(P_i);
   \}
4. C = KP^N \text{ mod } 128;
5. Write C;
\}

3.3.2 ALGORITHM FOR DECRYPTION

\{ 
1. Read n,N,K,C;
2. find \text{modinverse}(K);
3. PN = k^{-1}C \text{ mod } 128;
4. for i = N to 1 
   \{ 
   \text{decompose}(P_i);
   P_{i-1} = k^{-1}P^i \text{ mod } 128;
   \}
5. P = P0;
6. Write P;
\}

3.3.3 ALGORITHM FOR MODINVERSE

\{ 
1. read K,n;
2. find K_{ji}, \Delta; /* K_{ij} are the cofactors of the elements of K and \Delta is the determinant of K. */
3. find d such that (d\Delta) \text{ mod } 128 = 1; /* d is the multiplicative inverse of } \Delta. */
4. K^{-1} = (K_{ji}d) \text{ mod } 128;
\}

3.3.4 ALGORITHM FOR INTERLACE

\{ 
1. l = 1;
2. convert P into binary bits;
3. for i = 1 to n
\}
{ 
    for j = 1 to 28 
    { 
        temp(l) = b_{ij} ; 
        temp(l+1) = d_{ij} ; 
        l = l+2 ; 
    } 
} 

5. l = 1 ; 
6. for i = 1 to n { 
    for j = 1 to 28 { 
        b_{ij} = temp(l) ; 
        d_{ij} = temp(l+n*7) ; 
        l = l+1 ; 
    } 
} 

3.3.5 ALGORITHM FOR DECOMPOSITION 
{ 
1. l = 1 ; 
2. convert P into binary bits ; 
3. for i = 1 to n 
{ 
    for j = 1 to 28 
    { 
        temp(l) = b_{ij} ; 
        temp(l+n*7) = d_{ij} ; 
        l = l + 1 ; 
    } 
} 
4. l = 1 ; 
5. for i = 1 to n { 
    for j = 1 to 28 { 
        b_{ij} = temp(l) ; 
        d_{ij} = temp(l+1) ; 
        l = l + 2 ; 
    } 
} 
6. convert binary bits to decimal numbers ; 
}
3.4. ILLUSTRATION OF THE CIPHER

Consider the plaintext given below.

The policy of the other country is not clear. Let us watch for a few months and take a decision in respect of the external affairs and warfare. (3.4.1)

Let us focus our attention on the first sixty four characters given by:

The policy of the other country is not clear. Let us watch for a (3.4.2)

On writing the ASCII codes for characters in a columnwise manner, the above plaintext can be written in the form of a matrix given by

\[
P = \begin{pmatrix}
84 & 99 & 101 & 99 & 105 & 108 & 116 & 99 \\
104 & 121 & 32 & 111 & 115 & 101 & 32 & 104 \\
101 & 32 & 111 & 117 & 32 & 97 & 117 & 32 \\
32 & 111 & 116 & 110 & 110 & 115 & 115 & 102 \\
112 & 102 & 104 & 116 & 111 & 46 & 32 & 111 \\
111 & 32 & 101 & 114 & 116 & 32 & 119 & 114 \\
108 & 116 & 114 & 121 & 32 & 76 & 97 & 32 \\
105 & 104 & 32 & 32 & 99 & 101 & 116 & 97 \\
\end{pmatrix}
\] (3.4.3)

Here we take

\[
K = \begin{pmatrix}
53 & 62 & 24 & 33 & 49 & 18 & 17 & 43 \\
45 & 12 & 63 & 29 & 60 & 35 & 58 & 11 \\
8 & 41 & 46 & 30 & 48 & 32 & 5 & 51 \\
47 & 9 & 38 & 42 & 2 & 59 & 27 & 61 \\
57 & 20 & 6 & 31 & 16 & 26 & 22 & 25 \\
56 & 37 & 13 & 52 & 3 & 54 & 15 & 21 \\
36 & 40 & 44 & 10 & 19 & 39 & 55 & 4 \\
14 & 1 & 23 & 50 & 34 & 0 & 7 & 28 \\
\end{pmatrix}
\] (3.4.4)

On using the algorithm 3.3.1, we get

\[
P' = \begin{pmatrix}
57 & 14 & 121 & 40 & 109 & 45 & 122 & 3 \\
7 & 14 & 25 & 108 & 16 & 56 & 113 & 58 \\
21 & 87 & 12 & 6 & 0 & 18 & 33 & 119 \\
48 & 20 & 106 & 15 & 47 & 76 & 52 & 114 \\
97 & 112 & 81 & 33 & 18 & 9 & 63 & 2 \\
52 & 23 & 63 & 27 & 83 & 85 & 122 & 54 \\
61 & 40 & 105 & 37 & 93 & 74 & 18 & 39 \\
35 & 9 & 53 & 15 & 127 & 4 & 8 & 88 \\
\end{pmatrix}
\] (3.4.5)
On performing the interlacing mentioned in section 3.2, the new $P^1$ can be obtained in the form

$$
P^1 = \begin{pmatrix}
61 & 83 & 9 & 121 & 82 & 6 & 84 & 65 \\
127 & 70 & 17 & 5 & 78 & 87 & 16 & 6 \\
2 & 42 & 11 & 104 & 54 & 37 & 38 & 59 \\
42 & 3 & 91 & 100 & 63 & 110 & 15 & 30 \\
4 & 35 & 70 & 46 & 55 & 115 & 49 & 68 \\
9 & 33 & 42 & 61 & 83 & 6 & 24 & 55 \\
28 & 85 & 36 & 112 & 58 & 95 & 1 & 18 \\
91 & 24 & 43 & 46 & 20 & 98 & 65 & 106
\end{pmatrix}
$$

(3.4.6)

After carrying out all the sixteen rounds ($N = 16$), we get

$$
C = \begin{pmatrix}
115 & 35 & 112 & 78 & 96 & 21 & 25 & 88 \\
113 & 94 & 1 & 80 & 95 & 65 & 119 & 54 \\
53 & 22 & 49 & 67 & 108 & 99 & 35 & 90 \\
101 & 120 & 68 & 4 & 76 & 125 & 23 & 29 \\
31 & 60 & 122 & 90 & 86 & 41 & 95 & 16 \\
60 & 13 & 56 & 63 & 89 & 116 & 114 & 53 \\
0 & 77 & 30 & 68 & 106 & 53 & 30 & 70 \\
21 & 18 & 3 & 117 & 25 & 71 & 58 & 36
\end{pmatrix}
$$

(3.4.7)

The modular arithmetic inverse of $K$, denoted by $K^{-1}$, can be obtained as

$$
K^{-1} = \begin{pmatrix}
27 & 40 & 53 & 3 & 117 & 48 & 25 & 2 \\
41 & 60 & 17 & 92 & 5 & 21 & 106 & 81 \\
57 & 39 & 115 & 118 & 18 & 0 & 37 & 116 \\
94 & 97 & 52 & 27 & 94 & 102 & 104 & 19 \\
63 & 123 & 117 & 0 & 98 & 9 & 97 & 32 \\
61 & 50 & 54 & 60 & 101 & 12 & 69 & 56 \\
64 & 41 & 57 & 22 & 73 & 75 & 49 & 122 \\
71 & 61 & 17 & 32 & 42 & 88 & 81 & 113
\end{pmatrix}
$$

(3.4.8)

Here we are to note that we are able to obtain the $K^{-1}$ as the matrix of $K$ is nonsingular and the determinant of $K$ is relatively prime to 128. Further, it can be readily established that

$$
KK^{-1} \mod 128 = K^{-1}K \mod 128 = I.
$$

On taking the $C$ given in (3.4.7) and using the algorithm (3.3.2), we get
On performing decomposition, as we have mentioned in section 3.2, we get the new PN in the form

\[
\begin{pmatrix}
6 & 92 & 31 & 37 & 66 & 13 & 108 & 15 \\
100 & 10 & 54 & 59 & 104 & 82 & 119 & 47 \\
22 & 102 & 105 & 110 & 3 & 69 & 116 & 6 \\
79 & 108 & 38 & 113 & 40 & 53 & 26 & 55 \\
107 & 47 & 90 & 80 & 120 & 96 & 81 & 63 \\
118 & 112 & 40 & 42 & 42 & 79 & 18 & 86 \\
65 & 64 & 72 & 120 & 24 & 26 & 115 & 83 \\
114 & 48 & 35 & 58 & 54 & 57 & 91 & 66
\end{pmatrix}
\]

This process can be continued in the case of all the sixteen rounds (\(N = 16\)). Thus we get

\[
\begin{pmatrix}
18 & 60 & 83 & 55 & 30 & 51 & 64 & 85 \\
53 & 79 & 94 & 20 & 26 & 106 & 58 & 93 \\
79 & 96 & 116 & 7 & 115 & 60 & 92 & 96 \\
72 & 70 & 100 & 15 & 8 & 44 & 84 & 84 \\
66 & 83 & 65 & 127 & 19 & 99 & 108 & 83 \\
8 & 113 & 4 & 37 & 27 & 66 & 103 & 55 \\
100 & 111 & 3 & 33 & 104 & 7 & 123 & 30 \\
35 & 105 & 54 & 105 & 36 & 93 & 85 & 56
\end{pmatrix}
\]

This is the same as the plaintext given in (3.4.3).

Let us now consider another example, wherein, we have taken the complete plaintext, given by (3.4.1). This plaintext is containing 143 characters. To represent this in the form of a matrix consisting of \(n\) rows and \(m\) columns, where \(n = 8\) and \(m\) is having an appropriate value, we add one more character ($$ is added here) to the plaintext. With this
padding, the plaintext can be represented in terms of ASCII codes as follows:

\[
P = \begin{bmatrix}
\end{bmatrix}
\] (3.4.12)

Here, we perform the interlacing as we have mentioned earlier. Then, on adopting the process of encryption, we get the ciphertext, in hexadecimal notation, as shown below.

\[
80F0B933CC7C10376008E3C5F00DEB2DDDA2EDDE2558B0D1A6408A06B546C6EE1272F6D6
B46562281E77F64FB292826209AC928C32DF9D89A46A94D6549F7D9FD43D42DR2A6BC
4DE28B6319C4F13805748923A908151FC2EFBC271976300F989929EC84DF76579BD44CDBAC3
\] (3.4.13)

On using the process of decryption, we readily find that this ciphertext can be brought into the form of the original plaintext (3.4.1).

In what follows, we study the avalanche effect, and examine the strength of the cipher by considering the cryptanalysis.

### 3.5. AVALANCHE EFFECT:

Consider the plaintext given by (3.4.2). On applying the algorithm 3.3.1, the corresponding ciphertext, in its binary form, can be obtained as follows.

\[
111001101000111111110000110011001101100111001100111001110110101100001011010000111100011110101
0111110111000100000010000100000101101100110111110011011110100101110110101111101110101111010
1010110110011000100011001100111010011001001110010010110100110111111011010111010111010110101
1011010101001100101110110011011110110101110101110101111101110101101011101101101011111011011
10100010111011001101101011010110110110111011101110111011101110111110111011101110111011101111
010000110111110101101010110101101111101101011011011111011011011101110111011101110111111111111
\] (3.5.1)
On replacing the first character 'T', of the plaintext given in (3.4.2) by 'U', we get a change of one binary bit in the plaintext as the ASCII codes of T and U are 84 and 85.

On using the modified plaintext and applying the encryption algorithm, the ciphertext in the form

\[ \text{(3.5.2)} \]

Here we readily notice that (3.5.1) and (3.5.2), consisting of 448 bits, differ by 242 bits. This is quite significant.

Let us now change the key matrix element \( K_{33} \) from 46 to 47. These two also differ in one bit. Now we use the modified key and the original plaintext and apply the encryption algorithm. Thus, we get the ciphertext in the form

\[ \text{(3.5.3)} \]

On comparing (3.5.1) and (3.5.3), having 448 bits, we find that the ciphertexts differ by 235 bits. This departure is also considerable.

From the above analysis, we conclude that the avalanche effect is highly pronounced, and this shows that the length of the plaintext and
the length of the key are quite up to the mark. Further, we notice that
the confusion and diffusion arising in this cipher are quite significant.

3.6. CRYPTANALYSIS

In the literature of cryptography, the methods that are well known for
cryptanalysis are

1. Ciphertext only attack (Brute force attack)
2. Known plaintext attack
3. Chosen plaintext attack and
4. Chosen ciphertext attack.

Nevertheless, an algorithm is generally designed to withstand the
first two attacks[21]. In all these attacks, the algorithm and the
ciphertext are clearly known to the cryptanalyst.

In this analysis, the key matrix is of size nxn and each element in this
matrix is lying between zero and sixty three. Thus, the size of the key
space is $2^{6n^2}$.

Let us assume that the time required for the computation of the
encryption algorithm with one value of the key in the key space is $10^{-7}$
seconds. Thus the time required for the execution of the cipher with all
possible keys in the key space is

$$\frac{2^{6n^2} \times 10^{-7}}{365 \times 24 \times 60 \times 60} \approx 2^{6n^2} \times 3.12 \times 10^{-15} \approx 10^{1.8n^2} \times 3.12 \times 10^{-15} = 3.12 \times 10^{1.8n^2-15} \text{ years.}$$

From this, we notice that the time required for the brute force attack
is very large, when $n \geq 4$. When $n = 4$, it takes $3.12 \times 10^{13.8}$ years.
Let us now examine the known plaintext attack. In this, we know as many plaintext and ciphertext pairs as we require.

In this cipher, the basic relations describing the encryption are

\[ P^0 = P, \] \hspace{1cm} (3.6.1)
\[ P^i = (KP^{i-1}) \mod 128, \] \hspace{1cm} (3.6.2)
and
\[ P^i = \text{interlace}(P^i), \text{ i = 1 to 16.} \] \hspace{1cm} (3.6.3)

In order to have a compact representation, we replace the function interlace by \<>, and write (3.6.3) in the form
\[ P^i = <P^i>. \]

Now, on using the above equations, the relation connecting the ciphertext C and the plaintext P, at the end of the 16th iteration, can be written in the form
\[ C = (K(<(K...(K<(KP) \mod 128> \mod 128)> ...\mod 128>))\mod 128. \] \hspace{1cm} (3.6.4)

Here, the plaintext P occurring in (3.6.4) is subjected to the operation interlacing, and it cannot be taken out. Thus, P cannot be transferred to the left side, by taking the modular arithmetic inverse of P, as it could be done in the case of the classical Hill cipher. Thus, the expression containing only K cannot be determined in terms of C and P\(^{-1}\). Hence, this cipher cannot be broken by this attack too.

It may be mentioned here that the chosen plaintext attack and the chosen ciphertext attack are based upon intuitive ideas and they require
several choices of the plaintext or the ciphertext depending up on the attack. However, as it is pointed out by William stallings [21], here we restrict our attention to the first two cases only.

From the afore mentioned discussion, we conclude that it is difficult to break this cipher by the conventional cryptanalytic attacks.

### 3.7. COMPUTATIONS AND CONCLUSIONS

In this chapter, we have developed a block cipher by modifying the Hill cipher. In this, the key is represented in the form of a matrix of size nxn and the plaintext is represented in the form of a matrix of size nxm, where m≥n. Thus we are able to accommodate a large number of characters of the plaintext into the plaintext matrix.

In this cipher, we have adopted an iterative process. In each round of the iteration, we have performed multiplication of the plaintext matrix with the key matrix and modular arithmetic operation with 128. In every round, the modified plaintext is represented in terms of binary bits and these binary bits are interlaced so that we get a plaintext matrix of the same size. This sort of interlacing of the binary bits of the plaintext is expected to cause a lot of confusion in the structure of the plaintext before it becomes the ciphertext.

Here, we have taken the size of the key matrix as 384 bits. The size of the plaintext block is 448 bits in the first example and 1008 bits in the second example.

The cryptanalysis discussed in this chapter, clearly indicate that the
cipher is a strong one and it is difficult to break the cipher.

In the light of the above discussion, we find that the cipher under consideration can be applied to a plaintext of any size (with padding, if needed) and the strength of the cipher is quite significant as interlacing and multiplication with the key matrix, in different rounds of the iteration process, are causing a lot of confusion and diffusion in the elements of the plaintext before it attains the form of the ciphertext.