Implementation and Analysis of a Novel Block Cipher

M. Kiran Reddy
M.E. Communication systems,
SSN College of Engineering,
Chennai, Tamil Nadu, India

K. J. Jegadish Kumar
Assistant Professor, ECE Department, SSN College of Engineering,
Chennai, Tamil Nadu, India

ABSTRACT
This paper presents a new cryptographic technique for secured transmission of text and image data over communication networks. The new algorithm named Fixed Block with Dynamic key (FBDK) is designed using simple operations like XOR, circular shifts, substitutions. It uses input fixed blocks of any size based on the size of the original message bits and key size is dynamic for each block which makes it more resistant to cryptanalysis. The performance analysis of FBDK algorithm for digital images, text data is performed. Experimental tests are carried out in detail to show high quality, efficiency of FBDK algorithm.

Keywords
Cryptography, Ciphers, Encryption, Public Key, Random number generation

1. INTRODUCTION
Nowadays, network security is one of the major concerns in the modern world. In this regard, a strong security technique is required to protect user data. Cryptography techniques play an important role in secured transmission of data over communication network and ensure integrity, authenticity, confidentiality of information. Several encryption algorithms have been proposed like AES (Advanced Encryption Standard) [5], DES (Data Encryption Standard) [5] and RSA [5]. These algorithms provide very good encryption for text applications. However, these encryption schemes appear not to be ideal for image applications [8] due to bulk data capacity which require high computational time because of the large mathematical operations involved in algorithms like RSA.

The FBDK algorithm proposed in this paper do not use any mathematical operations like modular exponentiation. It is designed using simple operations like XOR [4], circular shifting [4] and substitutions [4]. It is capable of encrypting text and image data with good efficiency for plaintext and cipher text of any size. Since, the key size is varying for each block. This paper analyzes the encryption efficiency of FBDK block cipher along with its detailed performance analysis and quality. The rest of the paper is organized as: In Section 2, a detailed description on the design of the FBDK algorithm is given. In section 3, encryption efficiency and performance of the algorithm for text and image data is analyzed, and the last section concludes this paper.

2. PROPOSED ALGORITHM
As stated in Section 1, the FDBK algorithm is designed using simple operations like XOR operation, circular shifting and substitutions. To design the algorithm certain parameters are to be defined. These include block size (N bits), initial key for each block (K_I), key size for each block (K_B bits), number of shifts on initial key (n_1), number of substitutions on initial key (s_1), number of shifts on initial input message of size K_B bits (n_2), number of substitutions on initial message of size K_B bits (s_2), number of random digits (R), Index number (I). After, defining the variables the design of the encryption algorithm is as follows:

1. First generate a random sequence of N digits and obtain the input message sequence.
2. Generate K_I of size K_B such that $128< K_B<N-128$ and select K_B bits from message sequence which will be transmitted in first block.
3. Perform XOR between K_I and message sequence obtained in step-2.
4. Substitute and shift (K_I) s_1, n_1 times respectively and replace last sixteen digits with index number.
5. To the sequence obtained in step-3 append R random bits where $R=N-K_B$ to make it N bit block.
6. Next shift and substitute the sequence obtained in step-5 n_2, s_2 times which gives encrypted message.
7. Transmit the blocks obtained in step-4 and step-6 to the receiver and repeat steps 1-7 for all blocks.

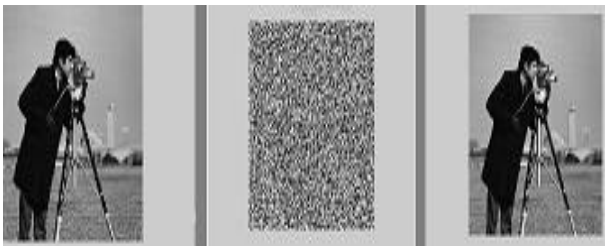
The decryption is just the reverse process of the encryption where the receiver decrypts message using the index number.

The number of circular shifts should be less than N because if we circular shift an N digit sequence N times then we get back the actual sequence and number of substitutions to be less than 2N times for the same reason. The substitutions, circular shifts and key size for each block are selected based on the message size and predefined random look up table and is known at transmitter and receiver side. As an example, Let N=1024, then $128<K_B<896$, $1<n_1, n_2<1024$, $1<s_1, s_2<2048$. The example look up table is shown in Table 1.

Table 1. Format for predefined random look up table

I	n_1	n_2	s_1	s_2	K_B
4506	789	1001	245	546	136
1234	215	589	896	754	512
5689	965	236	123	2002	768

To the receiver, only the index number is sent and based on this it will identify the parameters and decrypts the original message.



(a) (b) (c)

Fig 1 Encryption and Decryption of Image using FBDK with N=1024 bits, KB for first block is 256 bits. (a)Original Image, (b) Encrypted Image, (c) Decrypted Image

The figure 1 shows encryption of image file "Cameraman.tif" of size 256x256 with FBDK algorithm. The encryption time for this image is 18.254 Secs. The algorithm is fast enough to encrypt image files. The encrypted image is completely scrambled and the quality of encryption is measured by certain parameters that are discussed in section 3. When the input is text message, an important observation is made. The key size is random for each block and the block with minimum key size takes more time for encryption than the block with highest possible key size. Table 2 shows the variation of execution time with varying key size, fixed N and fixed input text message length for each block.

Table 2. Execution time for different KB with fixed N

N(bits)	KB(bits)	Message Length (bits)	Execution time (Secs)
1024	128	156568	0.042
1024	512	156568	0.035
1024	768	156568	0.028

From Table 2 it can be seen that with fixed block size (N) of 1024 bits and text message length of 156568 bits, different execution time for different key size was obtained. The least execution time (0.028 Secs) occurs when key size is maximum (768 bits). The key size is randomly taken by the encryption algorithm based on the look up table, and it is not possible to specify the key size for each block. But, the key size for initial block can be specified and thereafter for subsequent block a condition is used to select the key size. In the next section, the performance analysis and encryption quality of the algorithm is analyzed.

3. ENCRYPTION QUALITY AND PERFORMANCE ANALYSIS

A good encryption scheme must posses high encryption quality and low execution time [9]. In this section, encryption quality and performance analysis tests were conducted on FBDK algorithm using text and image files. The tests were performed using Microsoft Windows XP professional Version 2002 Service pack 3 on Intel(R) core(TM) Duo CPU, 1.83 GHz to 0.99 GB of RAM using Matlab 7.0.

3.1 Encryption Analysis

To determine the encryption quality two tests Viz., EQ measure, and PSNR (Peak Signal to Noise Ratio) value calculation is performed.

3.1.1 Encryption Quality (EQ) measure

When an image is encrypted there will be a change in its grey scale values. The EQ [8] measure gives the amount of deviation of grey scale values in original and encrypted images. Let I denote the original image and I' denote the encrypted image each of size M*N pixels with L grey levels. Let $H_L(I)$ denote the number of occurrences of each grey level L in the original image and $H_L(I')$ denote the number of occurrences of each grey level L in the encrypted image. The encryption quality [8] represents the average number of changes to each grey level L and is expressed mathematically as,

$$\text{Encryption Quality} = \frac{\sum_{L=0}^{255} \left| H_L(I') - H_L(I) \right|}{256}$$

The table 4 gives a comparison between GKSBC and FBDK in terms of EQ. All the images are of size 512X512 and FBDK has high EQ values compared to GKSBC [8]. Hence, the proposed algorithm proves to be efficient for image encryption when compared with GKSBC.

3.1.2 PSNR value

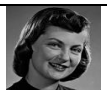
An efficient encryption algorithm produces an unintelligible image [3] as output. The parameter that can be used for determining the encryption efficiency of the algorithm is the PSNR (peak signal to noise ratio) [4] value in decibels (dB). Ideally, PSNR value between the actual image and encrypted image (PSNR_A_E) should be 0 dB and between the actual image and decrypted image (PSNR_A_D) it should be infinity [4]. Practically an efficient algorithm give a PSNR value less than 9 dB between actual image and encrypted image, a PSNR value between (30-50) dB between actual image and decrypted image. Table 3 shows the PSNR values obtained with various standard test images taken as input to the proposed algorithm.

Table 3. PSNR values for different Standard test images

Image name	Dimensions	PSNR_A_E (dB)	PSNR_A_D (dB)
Cameraman.tif	256 X 256	8.389	37.8532
Singer.jpg	225 X 225 X 3	8.6653	46.9137
Pirate.tif	512 X 512	8.9183	36.6502

The results in table 3 and table 4 show that the proposed algorithm provides efficient encryption and decryption of image files with a good encryption quality.

Table 4. EQ comparison between GKSBC and FBDK

Image file	original image	encrypted image	EQ (GKSBC)	EQ (FBDK)
Lena 512X512			663.82	714.65
Baboon 512X512			773.90	823.710
Girl 512X512			894.89	939.71

3.2 Performance Analysis

Encryption time [8] and throughput analysis has been done to determine the performance of the proposed algorithm. A good encryption algorithm should have low execution time along with high encryption quality. Encryption time refers to the time taken by algorithm to encrypt the data. Throughput indicates the speed of encryption that can be calculated from the encryption time. The table 5 gives a comparison between execution times of different encryption algorithm with text files of different sizes. The execution times for different algorithms excluding FBDK in table 5 are obtained from [8].

Table 5 Comparison of encryption time and throughput for different algorithm with FBDK

Input size (in Kbytes)	RC6 (Secs)	AES (Secs)	GKSBC (Secs)	FBDK (Secs)
1261	5.233	6.421	1.122	2.456
1357	5.822	7.847	1.234	2.678
1589	6.472	8.062	1.512	3.046
1605	7.349	8.793	1.557	3.329
Average Time (Secs)	6.219	7.780	1.356	2.877
Throughput (Megabytes/sec)	0.233	0.186	1.071	0.505

The experimental results in table 5 clearly indicates that the average encryption time for FBDK (2.877) and throughput (0.505 Mb/Sec) are much better when compared with RC6 and AES. GKSBC has highest throughput (1.071 Mb/s) but its encryption quality is less compared to FBDK. The results of various Encryption and performance analysis tests on text and image data show that the novel FBDK block cipher can be efficiently used for text as well as image data. The future work is to analyze the algorithm for its resistance to statistical attacks, brute force attack and test for its hardware compatibility.

4. CONCLUSION

This paper presents a design of FBDK algorithm and its encryption quality and performance analysis. The FBDK algorithm provides efficient encryption for both text and image data. The encryption time and throughput is also very high which makes it a good encryption algorithm. With images, the performance of the algorithm is very good with high encryption quality and peak signal to noise ratio values. The experimental test showed that FBDK algorithm outperformed various existing algorithms in efficiency and quality.

5. ACKNOWLEDGMENT

The authors wish to express heartful thanks to the faculty of ECE, SSN College of Engineering, Chennai for their support to this research.

6. REFERENCES

[1] Ashwak Al-Abiachi, M., Faudziah Ahmad, and Ku Ruhana. 2011. A Competitive Study of Cryptography Techniques over block cipher .

[2] Amitesh Singh Rajput, Nishchol Mishra, and Sanjeev Sharma. 2013. Towards the Growth of Image Encryption and Authentication Schemes.

[3] Chen, C.Y., Chang, C.C., and Yang, W.P. 1996. Hybrid method to modular exponentiation with precomputations.

[4] Morris Mano, M., and Michael ciletti, D. 2008. Digital Design. Fourth edition.

[5] Koblitz, N. 1994. A Course in Number Theory and Cryptography. Second Edition.

[6] Nitty Sarah Alex, Jani Anbarasi, L. 1985. Enhanced Image Secret Sharing via Error Diffusion in Halftone Visual Cryptography.

[7] Chen, G., Mao, Y., and Chui, C.K. 2004. A symmetric image encryption scheme based on 3D chaotic cat maps.

[8] Arul jothi, S., and Venkatesulu, M. 2012. Encryption Quality and Performance Analysis of GKSBC algorithm.