Dhaka Univ. J. Eng. & Tech. Vol. 1(1) 7-12, 2010 (July)

A New Approach of Symmetric Key Cryptography

Nadim Jahangir and Ahsan Raja Chowdhury[Ψ]

Department of Computer Science and Engineering, Northern University Bangladesh

[Ψ]*Department of Computer Science and Engineering, University of Dhaka.*

e-mail: **farhan717@cse.univdhaka.edu**

Abstract

In this paper, a new algorithm has been proposed for symmetric key polyalphabetic data encryption, namely Fourier Masking Encryption Algorithm (FMEA). It uses extended ASCII values to encrypt message of ASCII valued characters. The concept of Fourier series is exploited efficiently to generate the random key sequence from a given password, as randomness is the key property of polyalphabetic encryption. It has been shown that the FMEA cipher can provide higher-level cryptanalytic complexity for any unauthorized attempt to decryption. Experimental results are also provided here, which shows the efficiency of the proposed one over the existing algorithms.

Keywords: Cryptanalytic Complexity, Fourier Series, Masking Function, Polyalphabetic Encryption.

1. Introduction

Cryptography is the art of keeping message secret by different methods. There are several encryption algorithms that are the outcome of extensive research in the recent years. Some of them indeed provide good security but some others are vulnerable to either brute-force or cryptanalytic attack. Some of them are easy to implement in hardware with high processing power and storage capacity and some others are good for limited devices like mobile phones, PDAs etc. Most of the available cryptographic algorithms are based upon number theory which use finite-field such as **GF(*p*)** or **GF(2^n)**. Most of these number theoretic algorithms are more secure when they use large prime numbers or large binary words [1, 2]. But when the precision and bit-width increases, the hardware in which the algorithm is to be implemented must be sophisticated in processing power and storage capacity and hence it tends to high cost. An encryption/decryption algorithm has been developed, which exploits the versatility of Fourier series [3, 4]. It doesn't need any additional storage; rather provide a very good security. This can be useful in almost every system with memory and processing power constraints. It is a symmetric key polyalphabetic encryption technique where a single key is shared for both encryption and decryption algorithm. The symmetric key encryption method is depicted in Fig. 1.

The paper is organized as follows: Some basic definitions and literature review are discussed in Section 2. Proposed method is thoroughly discussed in Section 3 with Algorithm, Flow-Chart and example. Section 4 shows the Cryptanalytic Complexity of the Proposed Algorithm from different points of view. Proposed Algorithm is compared with the existing algorithms in Section 5. Section 6 presents the Conclusion of the paper and some important References are listed at the end of the paper.

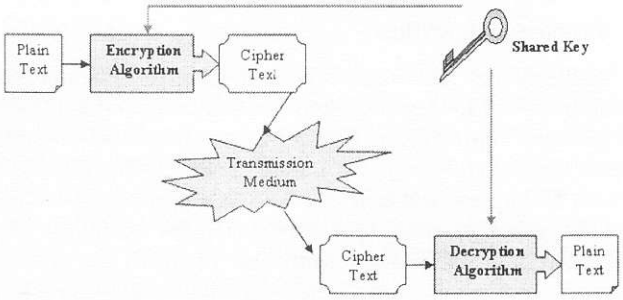


Fig. 1: Symmetric key encryption

2. Basic Definitions And Literature Review

In this section some basic definitions are given that are used throughout this paper. Some of the existing encryption algorithms are also discussed in this section.

A. Plaintext and Ciphertext: 'Plaintext' is the message that is to be encrypted and 'Ciphertext' is the encrypted message.

B. Polyalphabetic Encryption: 'Polyalphabetic Encryption' technique uses different keys for the encryption of individual characters in the plaintext. So in this scheme, several key values are required [1].

C. Diffusion and Confusion Property: Diffusion property of any encryption algorithm is responsible for relating one character in the plaintext with several characters in the Ciphertext so that the statistical relation between the plaintext and Ciphertext is reduced. Confusion property ensures that there is no or a little relation between the key and the Ciphertext [1].

D. Avalanche Effect: Catastrophic change in the Ciphertext for a little change in the key or in the plaintext is referred to as Avalanche effect [1].

There are several encryption algorithms for symmetric key encryption. There are also block cipher algorithms that operates on a single block at a time. But block ciphers can also be implemented for stream encryption in several modes [1]. For example, DES (Data Encryption Standard) is a block cipher technique but can be used for stream encryption in ECB (Electronic Code-Book), CFB (Cipher Feed Back), Counter modes. Other encryption algorithms

* Contact Author: Ahsan Raja Chowdhury, Lecturer, Dept of CSE, DU. e-mail: farhan717@cse.univdhaka.edu

are AES (Advanced Encryption Standard), BlowFish, RC4 etc. All of these algorithms are based upon number theory (finite-field is used). These algorithms use some additional storage like S-box (Substitution Box), Permutation Table, Initial Value etc. These algorithms are also complex in nature, as the encryption process is very much complex, and whereas complexity is a prerequisite for security. But when complexity is increased, the difficulties of hardware and software implementation are also raised. Computational complexity does not matter if the algorithm can provide with a strong defense against fraud. But if it is consider for developing any encryption algorithm for limited devices like mobile phones, PDAs, BlueTooth hardware etc, storage capacity and processing power of that device is also a matter of concern. In that case an algorithm will be explored; this is simpler in hardware implementation but holds a very strong position against intruders.

3. Proposed Algorithm

A mathematical function has been developed which is like Fourier series [3, 4] or trigonometric polynomial that is used to generate unpredictable key sequence. The function takes a password of any length chosen by the encryption party and results in key sequence. It is found that that by evaluating Fourier series, a good randomness can be achieved. The sequence is then XORed with the plaintext to get the Ciphertext. The function is the heart of this algorithm and named as 'Masking Function'. This function is presented in (1).

$$f(n,A)=\left\lfloor\left|A\frac{\sum_{i=1,3,5\ldots}^{M-1} P_i\cos\left(\frac{\pi(P_{i+1}+\pi)n}{500}\right)}{\sum_{i=1,3,5\ldots}^{M-1} P_i}\right|\right\rfloor \quad \ldots \quad (1)$$

where,

P = *password*

n = *1, 2, 3...length[plain-text]*

M = *length[P]*

A = *a sequence of plaintext*

Algorithm 1: Fourier_Masking_Encryption _Algorithm (FMEA)

Part – 1 (Constructing the Masking Function)

I. Choose any password in the form of ASCII text.

II. Convert the values of the characters in the chosen password to their equivalent extended ASCII values and put the values in the vector $P = (P_1, P_2, P_3 \ldots P_M)$. *[Here M = number of characters in the password]*

III. If M is odd then pad a one '1' at the end of the vector P. *[So that after this padding operation P_M becomes equal to '1' and the value of M is increased by one]*

IV. Using the definition of the (1) construct the masking function.

As the masking function is ready, encryption/decryption procedure is followed.

Part – 2 (Encryption/Decryption)

{Encryption}

I. Take the plaintext sequence in the form of ASCII text.

II. Convert the characters in the plaintext to their equivalent extended ASCII values and put the values in the vector $X = (X_1, X_2, X_3 \ldots X_N)$. *[Here N = number of characters in the plaintext]*

III. Generate the Ciphertext sequence $Y = (Y_1, Y_2, Y_3 \ldots Y_N)$ *by* $Y_i = X_i \oplus f(i, X_{i-1})$ where $X_0 = P_1$.

{Decryption}

I. Take the Ciphertext sequence in the form of ASCII text.

II. Convert the characters in the Ciphertext to their equivalent extended ASCII values and put the values in the vector $Y = (Y_1, Y_2, Y_3 \ldots Y_N)$. *[Here N = number of characters in the Ciphertext]*

III. Retrieve the plaintext sequence $X = (X_1, X_2, X_3 \ldots X_N)$ *by* $X_i = Y_i \oplus f(i, X_{i-1})$ where $X_0 = P_1$.

The encryption procedure is summarized in the flowchart of Fig. 2. To decrypt the Ciphertext, same algorithm is used except that the Ciphertext is given as input and Y is the retrieved plaintext in this regard.

Example 1: The algorithm is illustrated by an example which is summarized in Table I. In this example the plaintext and the password are X = STAY = (83, 84, 65, 89) and P = COMPLEX = (67, 79, 77, 80, 76, 69, 88) respectively. To make the length of P even, P = P || 1 = (67, 79, 77, 80, 76, 69, 88, 1) has been done. The masking function becomes,

f(n,A) = FLOOR(ABS((A*(67cos0.516n + 77cos0.522n + 76cos0.453n + 88cos0.026n))/308))

Table I: Example of FMEA encryption

X	A	n	$k = f(n, A)$	$Y = k \oplus X$	Y in ASCII Character
83	67	1	61	110	*n*
84	83	2	56	108	*l*
65	84	3	28	93	*]*
89	65	4	0	89	*Y*

4. Cryptanalytic Complexity Of The Proposed Algorithm

Cryptanalytic complexity of the proposed algorithm is provided by the following features:

- Randomness and unpredictability of Fourier series.
- One-to-many mapping between plaintext and Ciphertext.
- Diffusion, confusion and avalanche among plaintext, password and Ciphertext.

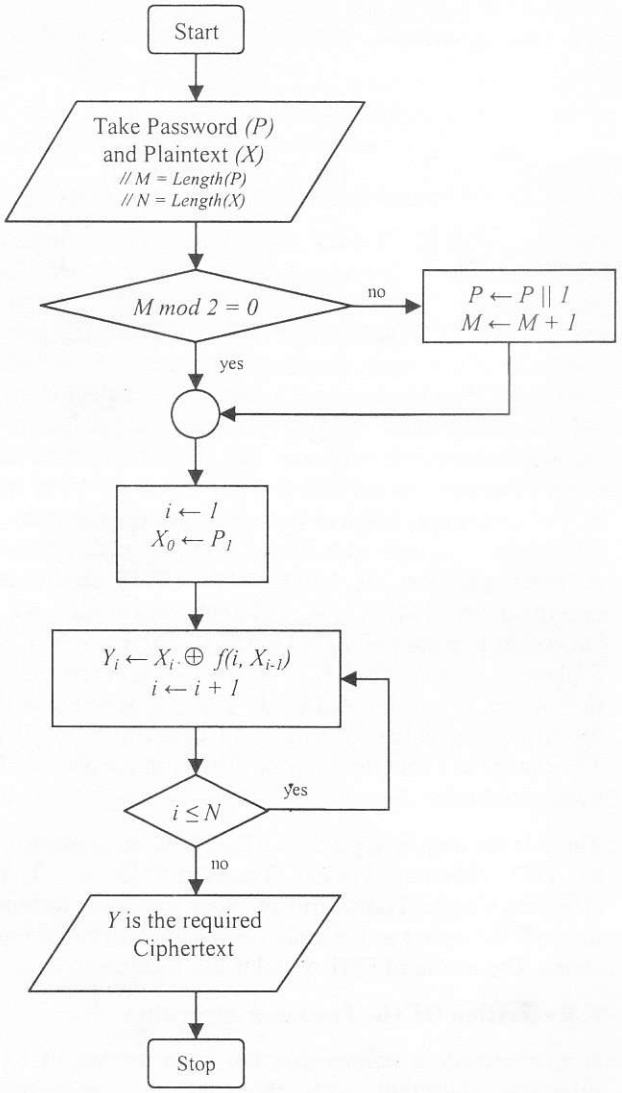


Fig. 2: Flow-chart for the proposed algorithm

A. Randomness and Unpredictability of Fourier Series

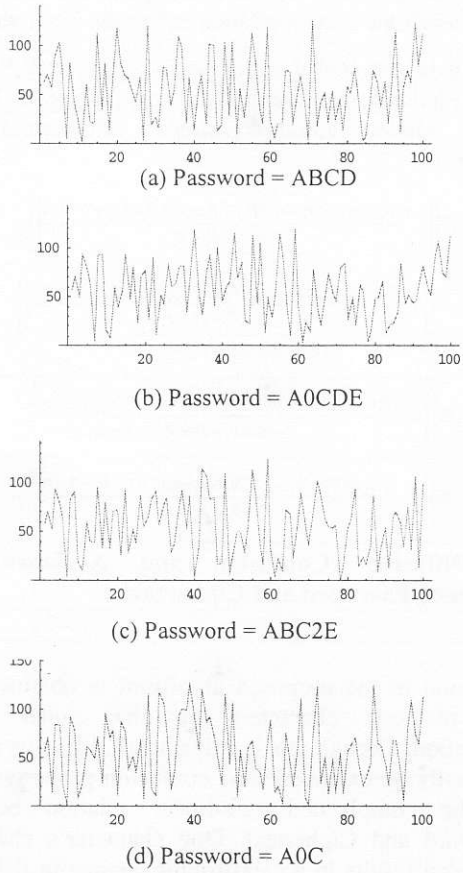
Fourier series is a trigonometric series [3, 4] that can be presented in its most general form as (2).

$$y = C + \sum_{n} A_n \cos \omega_n x + \sum_{n} B_n \sin \omega_n x \quad \ldots \quad (2)$$

This series can converge to any periodic function of 'x' if the values of the constants are chosen carefully. So it is said conversely that if we choose the values of the constants randomly, then unpredictable sequence can be obtained. This concept is the main motivation of proposed work work. The values of those constants from a given password have been extracted. The form of the original Fourier series as in (2) is not used in the proposed work, rather modified it as (1) to make it capable of generating unpredictable, mostly random and non-periodic key sequence. This modification is made by considering sampling, quantization and anti-aliasing techniques [5]. Randomness in key generation can be observed easily by Fig. 3. Here, generation of plaintext of length 100 ASCII characters and encrypted it by the proposed algorithm for different passwords. The graph in Fig. 3 shows the key sequence in each case.

B. One-to-many Mapping Between Plaintext and Ciphertext

The proposed algorithm maps the plaintext characters to a large space of Ciphertext characters. This feature thwarts any cryptanalyst in exploiting the statistical relation between plaintext and Ciphertext. For example the plaintext "AAAAAAAAAAAAAAAAAAAAAA" contains 24 As and when it is encrypted by FMEA with the password 2dÈú then the Ciphertext becomes Fq@iS|ExCgRrU□ChQi^}OqL`. Here it has been seen that the Ciphertext contains 23 different characters for a single plaintext character 'A'.



(a) Password = ABCD

(b) Password = A0CDE

(c) Password = ABC2E

(d) Password = A0C

Fig. 3: Randomness of the proposed algorithm in key generation

C. Sophisticated Solution Procedure in Finding the Password

By observing the proposed algorithm and the Masking Function it can infer that each Ciphertext character is a function of two plaintext characters and the password. The plaintext and Ciphertext characters can be equated as follows,

$$Y_1 = \textbf{\textit{Function}}(X_1, P_1, P_1, P_2, P_3 \ldots P_M)$$
$$Y_2 = \textbf{\textit{Function}}(X_2, X_1, P_1, P_2, P_3 \ldots P_M)$$
$$Y_3 = \textbf{\textit{Function}}(X_3, X_2, P_1, P_2, P_3 \ldots P_M)$$
$$\ldots \quad \ldots\ldots\ldots$$
$$\ldots \quad \ldots\ldots\ldots$$
$$Y_N = \textbf{\textit{Function}}(X_N, X_{N-1}, P_1, P_2, P_3 \ldots P_M)$$

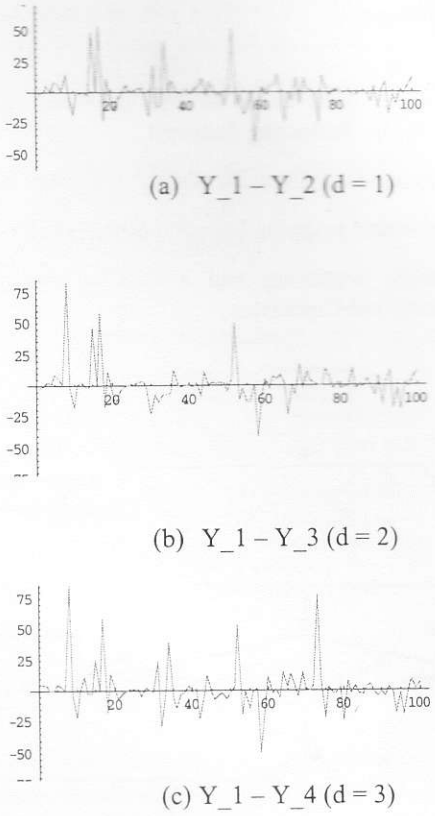
If Y and X are known (that is, if the attack **is known plaintext attack**), then M equations are required to solve for P. The above system of equation is a non-linear complicated equation with trigonometric polynomial. The password P is unknown and hence M is also unknown since it is the length of the password. So, solving the system is impractical. If however the value of M is known then there is also a difficulty in finding P if N<M. If M is known and N≥M then there is another complication, that is; the system cannot be solved if there are no sufficient plaintext and Ciphertext characters. One system of such equations is shown in (3).

Here, it is assumed that the password to be $P_1P_2P_3P_4$. This system holds a massive difficulty in finding P. There is no linear relation among plaintext, Ciphertext and the password.

$$Y_1 = X_1 \oplus \left| \left| P_1 \frac{P_1\cos(0.0063P_2 + 0.019) + P_3\cos(0.0063P_4 + 0.019)}{P_1 + P_2 + P_3 + P_4} \right| \right|$$
$$Y_2 = X_2 \oplus \left| \left| X_1 \frac{P_1\cos(0.0125P_2 + 0.039) + P_3\cos(0.0125P_4 + 0.039)}{P_1 + P_2 + P_3 + P_4} \right| \right|$$
$$\ldots \qquad (3)$$
$$Y_3 = X_3 \oplus \left| \left| X_2 \frac{P_1\cos(0.0188P_2 + 0.059) + P_3\cos(0.0188P_4 + 0.059)}{P_1 + P_2 + P_3 + P_4} \right| \right|$$
$$Y_4 = X_4 \oplus \left| \left| X_3 \frac{P_1\cos(0.0251P_2 + 0.079) + P_3\cos(0.0251P_4 + 0.079)}{P_1 + P_2 + P_3 + P_4} \right| \right|$$

D. Diffusion, Confusion and Avalanche Among Plaintext, Password and Ciphertext

Diffusion in the proposed algorithm is obtained by using two plaintext characters in one cipher character's generation. Though this is still a small diffusive approach, it can easily be extended. The confusion property is obtained by the complicated non-linear relation between the password and Ciphertext. One character's change in the password results in a catastrophic change in the Ciphertext which ensures a good avalanche effect. This changing effect is presented with the example of Fig. 4.



(a) Y_1 – Y_2 (d = 1)

(b) Y_1 – Y_3 (d = 2)

(c) Y_1 – Y_4 (d = 3)

Fig 4: Graph of Y_1 – Y_n with varying'd'

A fixed plaintext of length 100 ASCII characters is encrypted by FMEA for different passwords. All the passwords P_n are of length 16 ASCII characters. Y_n is the Ciphertext for password P_n. Number of different elements in P_n than P_1 is denoted by'd'. The graphs of Fig. 4 show the differences in the sequences of Ciphertext Y_n and Y_1. The change in Ciphertext for the change in the password has been presented.

There is no way to apply FFT (Fast Fourier Transform) [5, 6], DFT (Discrete Fourier Transform) [5, 6, 7], DCT (Discrete Cosine Transform) or other transform techniques to crack the cipher as the cipher is a highly distorted Fourier series. The nature of FMEA yields this distortion.

5. Evaluation Of The Proposed Algorithm

In this section, a comparison has been performed for the proposed algorithm with three of the most popular algorithms; AES [8], CMEA [9] and Vigenere Cipher [10]. The AES is an advanced standardization for block encryption based on finite-field $\mathbf{GF}(2^8)$. It uses a 128-bit key and performs on a 128-bit block. It has 10 rounds of computation and in each round substitution, shifting, mixing and round key adding operations are performed. It uses a S-box of 256 bytes and an Inverse S-box of 256 bytes. It also uses several stored data for performing encryption. The brute-force attack is impractical for AES as a key of 128 bit is used. It is cryptanalytically strong as well. Though it is a block encryption algorithm, it can also be extended for stream cipher. All things go well in the implementation of AES as users are able to use high processing power of

today's computers and related hardware. But if we consider an encryption algorithm for any limited devices like mobile phones and PDAs then , it is a matter of think about an efficient algorithm for these devices. So, additional storage which is required in AES must be reduced but not by the cost of security. There is an algorithm namely CMEA (Cellular Message Encryption Algorithm) [9] used for cellular phone message encryption (like numbers dialed). It is pretty good one but also uses a Cave Table which contains 256 bytes of previously stored data. Recent works [9] have been able to crack CMEA cipher. So CMEA will be no longer strong. The proposed algorithm does not need any kind of additional storage but can provide with good security as AES and CMEA. AES and CMEA are complex in both software and hardware implementation. The proposed technique is simple as:

- Only M/2 trigonometric calculations are required in each cipher character generation.
- (3M/2)+1 multiplications are performed in each cycle.

A comparison is summarized in Table II. By FMEA a pretty good one-to-many mapping between plaintext and Ciphertext characters can be obtained. One-to-many mapping is needed to reduce the possibility of cracking any cipher by observing its frequency description. The best known polyalphabetic cipher 'Vigenere cipher' [10] is good to provide on this need but it still has some limitations. For example: if any analyst can somehow determine the length of the keyword that is used to encrypt any plaintext which has enough length and good number of repetitions of characters, then Vigenere cipher may also have repetitions in its cipher.

Example 2: In this example, a plaintext is encrypted using Vigenere Cipher algorithm (for keyword DECEPTIVE) and repetitions of characters are shown using shaded cells in Table III. In this case, VTW is repeated twice in the ciphertext for the same plaintext repetition of RED and for this the portion of the key is EPT which is also repeated. This repetition can be avoided if the keyword's length is large enough. This repetition in the Vigenere cipher aids an analyst in predicting the length of the keyword. When the length is known, the keyword can be found by observing the frequency description of the cipher.

These limitations in Vigenere cipher can be overcome by using FMEA. The reason is that: the polyalphabetic round key generated by FMEA is highly random in nature of sequence. FMEA is made as it can generate highly non-periodic and unpredictable round key sequence. Moreover this sequence is dependent not only on the password self but also on the plaintext. So it can provide a highly secure key sequence.

Example 3: The plaintext in the Example 2 is encrypted with password DECEPTIVE using FMEA and presented in Table IV.

It is seen that there is no repetition in the FMEA Ciphertext for the repetition of RED in the plaintext. FMEA key sequence is a pretty good random sequence that can also be used in other cryptographic applications than only encryption. For example this can be used to produce pseudorandom numbers and pseudorandom number is a very important tool in cryptographic application like authentication. In the classical pseudorandom number generators, there had been used modular arithmetic with large prime numbers. As FMEA doesn't use modular arithmetic but still well in generating random sequence, it can easily be used in such applications where computational complexity must be reduced. Notably in the algorithm for the authentication technique proposed in the paper [11] FMEA can be used effectively.

Table II: Comparison between the proposed algorithm with AES and CMEA

Feature	AES	CMEA	Proposed One
Key/Password Size (bit)	128	64	8n where, n is the length of the password which can be of any value
Plaintext Size (bit)	128	2-6	8
Use Additional Storage	Yes	Yes	No
Method	Number Theoretic in $GF(2^8)$	Number Theoretic	Trigonometric
Use Several Rounds	Yes	Yes	No
Possible Keys/Passwords	2^{128}	2^{64}	$S = \sum_{k=1}^{M} 256^k$ Theoretically when, $M \rightarrow \infty$, $S \rightarrow \infty$

Table III: Ciphertext for a given plaintext in Vigenere Cipher (for keyword DECEPTIVE)

Key	D	E	C	E	P	T	I	V	E	D	E	C	E	P	T	I	V	E	D	E	C	E	P	T	I	V	E
Plaintext	W	E	A	R	E	D	I	S	C	O	V	E	R	E	D	S	A	V	E	Y	O	U	R	S	E	L	F
Ciphertext	Z	I	C	V	T	W	Q	N	G	R	Z	G	V	T	W	A	V	Z	H	C	Q	Y	G	L	M	G	J

Table IV: Ciphertext for a given plaintext in FMEA (for keyword DECEPTIVE)

Plaintext	W	E	A	R	E	D	I	S	C	O	V	E	R	E	D	S	A	V	E	Y	O	U	R	S	E	L	F
Ciphertext	j	p	Q	X	a	c	k	@	E	S	a	□	d	t	R	Q	Q	E	]	S	L	G	M	p	d	Y	I

6. Conclusion

This paper proposes a novel idea of expanding Fourier series for Polyalphabetic Symmetric Key Encryption. The proposed algorithm, namely Fourier Masking Encryption Algorithm (FMEA) uses the versatility of Fourier series, which is a trigonometric polynomial, rather than using modulo arithmetic. The algorithm not only possesses one-to-many mapping between plaintext and Ciphertext, but also establishes diffusion, confusion and avalanche among plaintext, password and Ciphertext. The proposed algorithm is also easy to implement in hardware, mobile phones, PDA as it doesn't require any extra storage capacity. The efficiency of the proposed algorithm over the existing algorithms is shown in experimental results which show the superiority of the proposed algorithm.

References

[1] Stallings. W, 2003, *Cryptography & Network Security (Principles & Applications)*, 3rd Edition.

[2] Telang. S.G, 1996, *Number Theory*, First Edition.

[3] Web-link: http://mathworld.wolfram.com/FourierSeries.html (last accessed on 11-01-2010)

[4] Web-link: http://en.wikipedia.org/wiki/Fourier_series (last accessed on 11-01-2010)

[5] Proakis. J. G, Manolakis. D. G, 2003, *Digital Signal Processing (Principles, Algorithms & Applications)*, 3rd Edition, Prentice Hall India.

[6] Haykin. S, Veen. B. V, 1999, *Signals and Systems*, John Wiley & Sons, Inc.

[7] Winograd. S, 2002"On Computing the Discrete Fourier Transform", Math. Comp., Vol. 32, pp. 177-199.

[8] Nechvatal. J., 2002, et al. "Report on the Development of the Advanced Encryption Standard", *National Institute of Standards and Technology*, October 2

[9] Wagner. D, Schneier. B, Kelsey. J, 1997, "Cryptanalysis of the Cellular Message Encryption Algorithm. Advances in Cryptology", *in proceedings, CRYPTO 1997*, Vol. 1294, Lecture Notes in Computer Science, Springer-Verlog, pp. 526–537.

[10] Web-link: http://www.trincoll.edu/depts/cpsc/cryptography/vigenere.html (last accessed on 11-01-2010)

[11] Rahman. M. L., Rafique. S., Jabiullah. M. I., Rahman. S. M. M, 2006, Strong Authentication Using Pseudo-Random Bit Stream, *Dhaka University Journal of Science Vol.* 54(1), pp. 59-61.