Attacking of stream Cipher Systems Using a Genetic Algorithm

Ali A. Abd [*] ***, Hameed A. Younis*** [**] ***, and Wasan S. Awad*** [***]

* *Dept. of Computer Engg., College of Engg., University of Basra.*
***Dept. of Computer Science, College of Science, University of Basra.*
*** *Dept. of Computer Science & Info. Systems, Technology University.*

Abstract

Stream cipher is one of the hard electronic cipher systems because of high security and difficulty in breaking it. This work is considered a new approach to cryptanalysis based on the application of direct search algorithm called genetic algorithm. It concentrates on showing the applicability of genetic algorithm concepts as a powerful tool in breaking cryptographic systems. The cryptanalysis is based on attacking stream cipher systems by finding the equivalent linear system.

The goal of the genetic algorithm is finding the shortest linear feedback shift register that generates the known key stream through finding the initial state, feedback polynomial, and the shift register length.

Previously, there were methods to find the linear equivalence by using Massey algorithm and neural networks. Now, the proposed method is considered the first attempt to find it via genetic algorithm with variable chromosomes lengths within genetic population leading to minimized average number of generations and accordingly less computational time.

The proposed artificial system has been applied successfully to break a number of linear and nonlinear stream cipher systems, such as Hadmard system and Bruer system.

Introduction

Attacking cipher systems can be done by using a number of methods which can be classified into different classes according to the available information:

- **Cipher-text only attack:** The cryptanalyst has the cipher text of several messages, all of which have been encrypted using the same encryption algorithm.
- **Known-plaintext attack:** The cryptanalyst has access not only to the cipher text of several messages, but also part of the plaintext of these messages.
- **Chosen-plaintext attack:** The cryptanalyst not only has access to the cipher text and associated plaintext for several messages, but also chooses the plaintext that gets encrypted.
- **Adaptive–chosen plaintext attack:** This is a special case of a chosen-plaintext attack. Not only can the cryptanalyst choose the plaintext that is encrypted, but he can also modify his choice based on the results of previous encryption.

In this paper, the adopted method is the known-plain text attack. In this method, cipher text and part of the plain text are known [1]. The security of the modern cryptography is based on the key (K).

This paper presents a complete genetic algorithm (GA) to find the linear equivalence of a given key stream through finding: Initial state, feedback function, and shift register length.

Stream Cipher Systems

Stream cipher systems convert a plain text to a cipher text one bit at time. A key stream generator outputs a stream of bits: $K_1, K_2 \ldots\ldots K_i$. This key stream is XORed with a stream of plain text bits: $P_1 \ldots\ldots P_i$ to produce a stream of cipher text bits $C_1 \ldots\ldots C_i$.

At the decryption end, the cipher text bits are XORed with an identical key stream to recover the plain text bits as shown in Fig.(1)[2]:

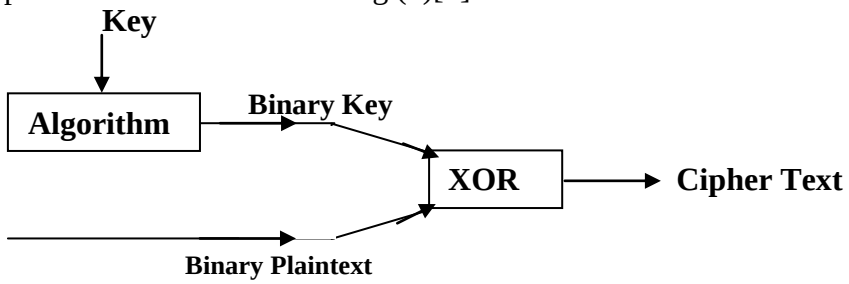

Fig.(1): Stream Cipher Encryption System

The main component of the key stream generator is the linear feedback shift register (LFSR) which consists of two parts: shift register and feedback function as shown in Fig.(2).

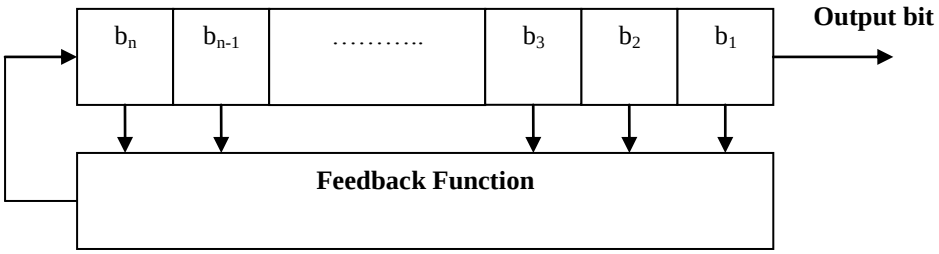

Fig.(2): Linear Feedback Shift Register (LFSR)

Each time a bit is needed; all the bits in the shift register are shifted to the right. The new left-most bit is computed as a function of the other bits [2]. The period of the shift register is the length of the output sequence before it starts repeating.

Linear Equivalence

A binary sequence may be generated by using more than one LFSR. The shortest LFSR generating that sequence is called the linear equivalence. The characteristics polynomial for this LFSR is called the minimal polynomial with degree equal to the linear equivalence. Furthermore, the number of bits for the LFSR is equal to that of the linear equivalence of the sequence generated from that register.

Attacking Methods

The attacking methods of stream cipher systems can be classified, based on the information processing approaches, into two major parts: Classical methods, and Modern methods. The classical methods include: Matrix method, Berlekamp-Massey algorithm, and Correlation and fast correlation methods. The modern methods include: Neural network method and Genetic algorithm method.

In matrix method, if n is the linear equivalence of the sequence of keystream, then we must know at least (2n) consecutive bits of the sequence (keystream) [3], while the iterative

algorithm introduced by Berlekamp-Massey is the shortest linear feedback shift register, capable of generating a finite sequence of bits [3].

Siegenthaler (1985) demonstrated a method that LFSRi part of key can be found independently of the other LFSRs parts, by using the "divide and conque" technique. Meier and Staffelbach (1988) developed two algorithms (A and B), which are much faster than the above attack (Siegenthaler).

On the other hand, the modern methods depend on different approaches for information processing such as biological-like processing.

Abbas N.M. [4] showed the applicability of multilayer neural networks with back propagation to crack a linear stream cipher with LFSR as a key generator, assuming that the cryptanalyst has obtained a finite sequence of known plaintext. Spillman et al [5] showed the use of genetic algorithm in the cryptanalysis of simple substitution ciphers. Mattews et al [6] demonstrated the use of genetic algorithms to break classical transposition ciphers by finding the transposition sequence used. A-Ageelee S.A. [7] developed a correlation attack by using genetic algorithm to reduce the number of attempts.

Application of the Proposed Algorithm to the Key Generation

In this work the main goal of using GA is to find the equivalent LFSR which leads to get the feedback polynomial, the initial state, and the length of the LFSR, knowing part of the plain text.

The first step is started with getting the key representation. The key is represented as a binary string (binary chromosome). This chromosome should be variable lengths and have an even length Lx. The chromosome is divided into two parts one for the feedback function and the other for the initial state of the LFSR equivalent to the attacked generator as shown in Fig.(3).

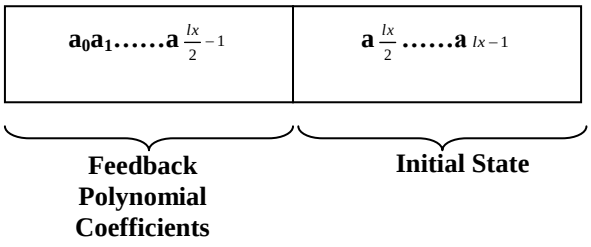

Fig.(3): Chromosome Structure in the GA population

The second step is to get the fitness function that evaluates the fitness of each string in the population. The fitness function adopted in this work may be:

$$F(x)=(2*(L_s-e))/ L_x$$

Where:

F(x): is the adopted fitness function.

L_s: is the length of the input (known or given) keystream.

e: is the difference between the generated sequence and the given sequence.

L_x: Chromosome length.

The following parameters have been chosen for the adopted genetic algorithm:

1. Selection strategy: Roulette wheel selection.
2. Crossover: 2-point crossover.
3. Replacement Policy: Both parents replacement.
4. Termination condition: e=0(i.e. $F(x)=2L_s/L_x$) or maximum number of generations is reached.

Fig.(4) shows a complete flow chart that represents the proposed genetic algorithm adopted for the stream cipher attacking.

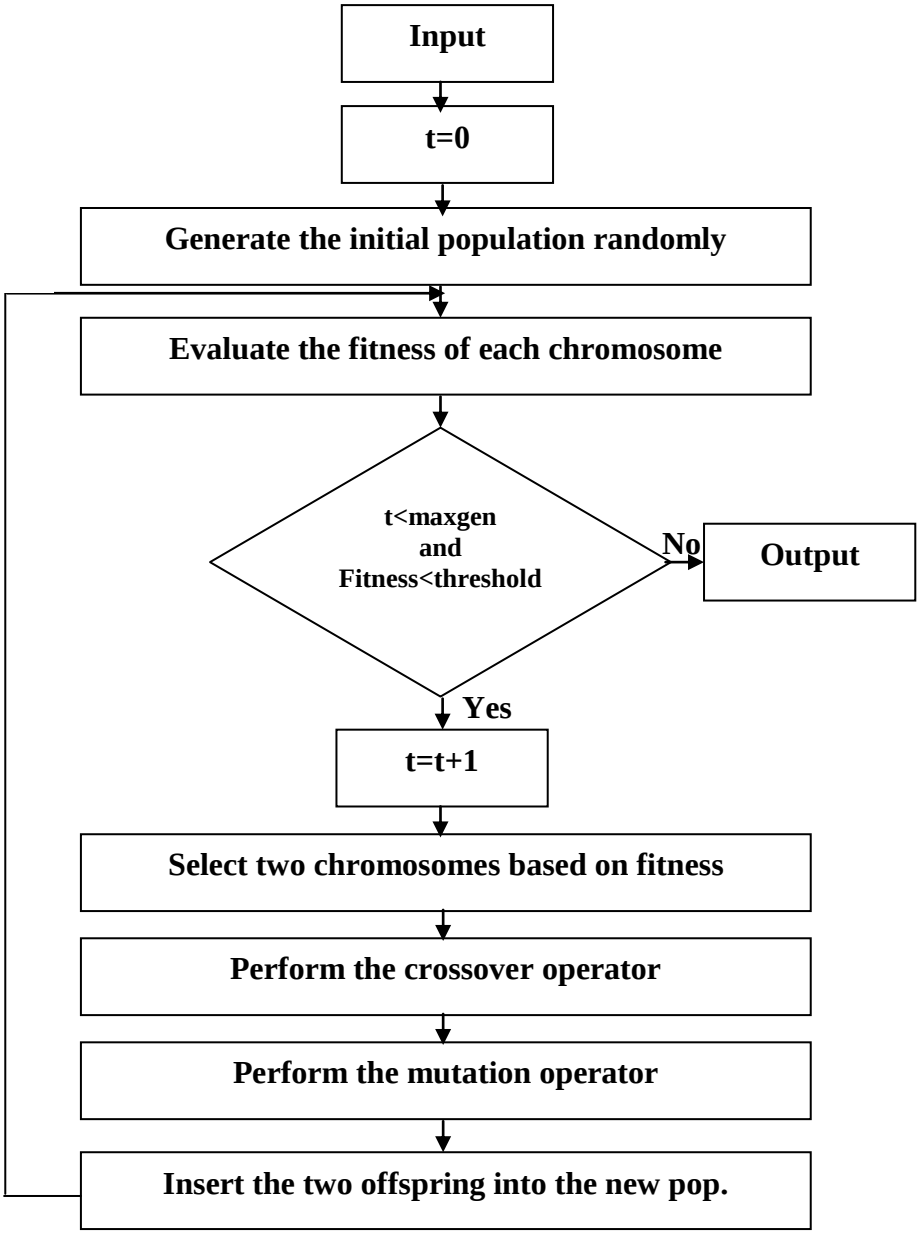

Fig.(4): Flow chart of the proposed genetic algorithm

Experimental Results

The simulation of the proposed algorithm is programmed. It is applied to sequences, which are created using stream cipher systems whose combining functions are linear feedback functions and nonlinear feedback functions (Hadmard and Bruer systems).

Experiment 1: This experiment is done to select the optimum values for crossover and mutation probabilities. The results are shown in Table (1) and Table (2) below:

Table (1): Optimum values for P_c

P_c	Average number of generations
0.3	26
0.5	29
0.7	23
0.8	25
0.9	**22**
1.0	24

Table (2): Optimum values for P_m

P_m	Average number of generations
0.0001	21
0.001	**17**
0.03	21
0.05	22
0.09	24
0.1	23
0.2	25
0.3	23

From the above results we found that the crossover rate for a minimum average number of generations is 0.9 and the mutation rate is 0.001. Fig.(5) below shows these results graphically.

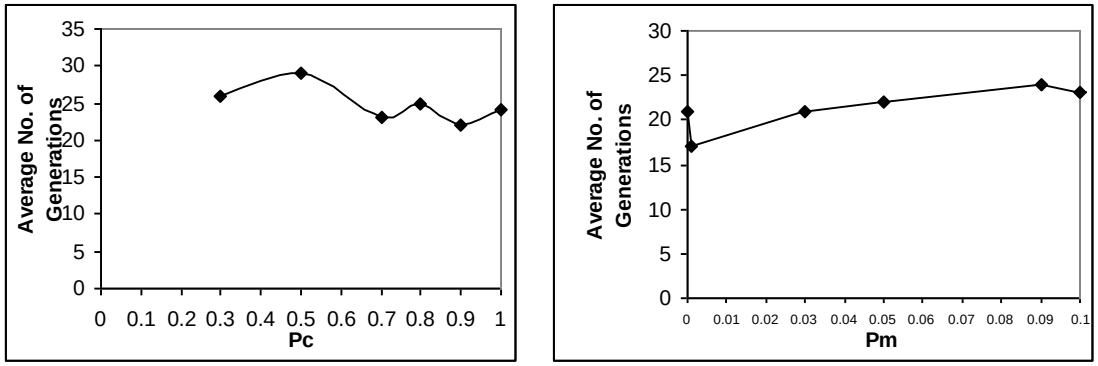

Fig.(5): Results of Crossover and Mutation Probabilities

Experiment 2: In this experiment, a linear stream cipher is chosen with different LFSR lengths and the corresponding average number of generations. The results are shown in Table (3) below:

Table (3): Results for different LFSR lengths

LFSR length	Known key stream length (2n)	Average number of Generations
5	10	22
6	12	53
7	14	21
8	16	97

From the above results, we conclude that whenever the length of the shift register (key size) increases, the average number of generations increases too.

Experiment 3: This experiment is for examining the effect of the known key stream length on the required number of generations. Linear stream cipher with shift register of length 5 bits is chosen. Table (4) below gives the results, in which we note that the average number of generations is zero when the length of the known key stream is less than twice the value of the linear equivalence. We may note also that whenever the length of the known key stream increases, the average number of generations decreases leading to a shortest computational time for the GA.

Table (4): Results of Experiment 3

LFSR length	Key stream length (2n)	Known key stream length	Average number of generations
5	10	5	0
5	10	10	67
5	10	15	53
5	10	20	42
5	10	30	35

Experiment 4: In this experiment, two nonlinear stream cipher systems, which are Hadmard and Bruer systems, are adopted. The obtained results are summarized in Table (5) below.

Table (5): Results of Experiment 5

System name	LFSR length	Known key stream length	Average number of generations
Hadmard	2,3	12	85
Bruer	2,3	12	98

The average number of generations of Bruer system is greater than that of the Hadmard system because the nonlinearity degree of Bruer system is greater. Furthermore, we note that the nonlinear systems need more time to be broken than the linear systems because the nonlinearity degree of the key generator is greater.

Conclusions

The work of this paper has developed a genetic algorithm for breaking stream cipher systems with a known plaintext attack. This algorithm should find the shortest LFSR which generates a sequence of key stream knowing part of it. The proposed system requires less computational time and information compared to the previous works, in which GA is used to reduce the number of trails when treating with nonlinear systems. A final conclusion is that the nonlinear stream cipher systems need more time for breaking it than the linear systems because the degree of nonlinearity is greater.

References

[1] D.E.R. Denning, "Cryptography and Data Security", Addision-Wesley Publishing Company Inc., USA, 1982.

[2] B. Schneider, "Applied Cryptography, Protocols, algorithms, and Source Code in C", John Wiley and Sons Inc., USA, 1996.

[3] W. Stalling, "Cryptography and Network Security", Pearson edition, Inc., USA, 2003.

[4] Abbas N.M., "Attacking Stream Cipher System Using Networks", M.Sc. Thesis, Technology University, Baghdad, 1996.

[5] Spillman R., et al, "Use of a Genetic Algorithms in the Cryptanalysis of Simple Substitution Ciphers", Cryptologia, Vol. XVII, No.1, PP.31-44, Jan. 1993.

[6] Mattews R. A. J., "The Use of Genetic Algorithms in Cryptanalysis" , Cryptologia, Vol. XVII, No.2, pp. 187-201, April 1993.

[7] Al-Ageelee S.A., "Use of Genetic Algorithm in the Cryptanalysis of Stream Cipher Systems", Ph.D. Thesis, Technology University, Baghdad, 1998.