JP Journal of Algebra, Number Theory and Applications
© 2017 Pushpa Publishing House, Allahabad, India
http://www.pphmj.com
http://dx.doi.org/10.17654/NT039050785
Volume 39, Number 5, 2017, Pages 785-802 **ISSN: 0972-5555**

AFFINE CIPHER CRYPTANALYSIS USING GENETIC ALGORITHMS

Yaqeen S. Mezaal[1] and Seevan F. Abdulkareem[2]

[1]Medical Instrumentation Engineering Department
Al-Esraa University College
Baghdad, Iraq
e-mail: yakeen_sbah@yahoo.com

[2]Computer Engineering Techniques Department
Al-Mansur University College
Baghdad, Iraq
e-mail: eng_seevan85@yahoo.com

Abstract

Genetic algorithms (GAs) have been used as a powerful tool for cryptanalyzing affine ciphers in this paper for the first time. They are one of heuristic search techniques which use natural selection. They select the optimal solution by using selection, crossover and mutation operations. The useful parameters in GAs are kept in the memory and the best values of fitness have been selected to represent the next generation. The frequencies of single letter have been used as an essential factor in the fitness function of the adopted GAs operations for affine cryptanalysis. By this tool, a high number of letters have been recovered to discover a plaintext of 375 letters by a fitness value of 95% at 120 generations in less than three minutes as compared to classical affine cryptanalysis without using GAs.

Received: April 20, 2017; Revised: May 29, 2017; Accepted: June 9, 2017

2010 Mathematics Subject Classification: 94A60.

Keywords and phrases: affine cipher, genetic algorithms (GAs), cryptanalysis, fitness value.

1. Introduction

Cryptography represents the concealment and privacy in writing. The main goal of cryptography is to provide an inexplicable message to an unauthorized reader. The cryptanalysis term means some procedural steps to recover the plaintext and/or key from a ciphertext. In general, cryptanalysis can be defined as the searching process for weakness points in the design of cryptosystems (or ciphers). A distinctive message takes an understandable text (known as the plaintext) and some secreted information (known as the key) as its input and generates an encrypted version of the original message (known as the ciphertext). An attacker on a cipher can exploit the ciphertext alone, or it can use some amount of both plaintext and its corresponding ciphertext. For example, in the brute force attack, the attacker attempts each feasible key on a portion of ciphertext until a comprehensive transformation into plaintext is acquired. However, it has the drawbacks of huge computational difficulty and long processing time. For such these disadvantages, the optimization heuristic techniques like genetic algorithms (GAs) have been used in the cryptanalysis of ciphers. These techniques are increasingly requested since they eliminate the required time by human interaction with a search process. The use of GAs has the possibilities to conduct a directed random search of a key space and deduce the key size [1, 2].

The pioneers in the use of GAs in the cryptanalysis of ciphers are Spillman et al. in 1993 [3]. They presented a new method of cryptanalysis to discover the key for a simple substitution cipher based on directed random search GAs. In the same year, Matthews used GAs as an influential tool in the breaking of cryptographic systems based on transposition ciphers [4]. He explained that GAs can significantly enhance cryptanalysis by powerfully searching large key space. In [5], GAs have been used to cryptanalyze polyalphabetic substitution cipher. The possibility of GAs for searching the key space of encryption scheme has been investigated. The results in the implementation of polyalphabetic substitution cipher cryptanalysis have shown the considerable influence of ciphertext size on recovered plaintext letters. The elapsed time for cipher cryptanalysis has been decreased by using GAs. In 2010, Turcinhodzic examined monoalphabetic substitution ciphers (shift transformations and affine transformations) and the approaches of their decrypting using the statistical features of natural language. A creative program for encryption and decryption based on affine transformations has been submitted [6]. In 2011, the cryptanalysis of the Vigenère cipher by using GAs has been presented. The applicability of GAs for searching the key space of encryption scheme has been investigated using frequency analysis as an important factor in the objective function [7]. Using GAs to crack Vigenère cipher has been found to be a competent method of cryptanalysis based on the feature of comparing the letter frequency rate in the model of text. Memetic and genetic algorithms have been implemented to break Hill ciphers using MATLAB simulator as stated in [8]. Different parameters were tested using GAs for Hill cipher cryptanalysis such as population and the required time for different number of generations. In 2014, GAs have been used to attack a simple cryptographic cipher, called *monoalphabetic substitution* as reported in [9]. The effect of different mutation rates for different population sizes to discover the key for a monoalphabetic substitution cipher has been studied and analyzed. In [10], GA has been used to decipher an Arabic encrypted text by Vigenère cipher. The occurrence frequency of Arabic letters is determined by using the texts of the holy book of Quran, since it has affluent language characteristics compared to other well-known books. GA has been investigated to discover the key letters for diverse key lengths and ciphertext sizes.

In this paper, a parametric study to cryptanalyze affine ciphers using GAs has been presented. A high percentage of letters have been discovered to recover a plaintext by a fitness value within a short time. The frequencies of single letter have been used as an essential factor in the fitness equation of GAs. It is obvious from the proposed tool to cryptanalyze affine ciphers that in a short run, it will reach close enough to the correct plaintext that a visual inspection of the resulting plaintext could be used to conclude any misplaced letters. This is theoretically very interesting outcome in the cryptanalysis applications as compared to classical affine cryptanalysis.

2. Affine Cipher

Affine cipher is a kind of substitution cipher, in which each letter in the alphabet is converted to its numeric equivalent, encrypted by a simple arithmetical equation and converted back to the letter. Namely, each letter encrypts to another letter, and resumes another time since the cipher is essentially a standard substitution cipher with a specific rule. Affine ciphers can be easily made a system noticeably secure by multiplying each plaintext value by a different number and then inserting a shift [11].

The word "affine" is applied for mathematical transformations that maintain a "kinship" between the original object and the transformed object. For example, points that are close together should be transformed into points that are also close together. "Affine" comes from the same root word as "affinity" [11].

After the Caesar cipher, the easiest method of enciphering transformation is an affine transformation, which multiplies each plaintext value by a different number and then adds a shift value.

The encryption function can be expressed by [11]:

$$Y = (mQ + b) \bmod n, \tag{1}$$

where n is the plaintext size, m is the multiplier value and b is the shift magnitude. The multiplier m in the equation above should be relatively prime to n, otherwise decryption is impossible. With the aim of decryption, we must solve equation (1) for Q. A distinctive solution can be found only if an inverse of m modulo n exists. The inverse of m can be simply determined by the extended Euclidean algorithm, and consequently, we can have the deciphering transformation (decryption function) by [11]:

$$Q = m^{-1}(Y - b)(\bmod n), \tag{2}$$

where m^{-1} is the modular multiplicative inverse of a modulo n that satisfies the following equation:

$$1 = mm^{-1} \bmod n. \tag{3}$$

For example, to encipher WAR LOST, let us use affine transformation with the ordinary alphabet. Use 7 as the multiplier and 10 as the shift. Then recover the plaintext. The ordinary alphabet associations are explained in Table 1.

Table 1. The ordinary alphabet associations with their numerical equivalent

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

From Table 1, the letters of the plaintext message can be transformed to their numerical equivalents as follows:

22 0 17 11 14 18 19.

Then we can calculate encryption elements as follows:

$$Y \equiv 7Q + 10 \equiv 7 \cdot 22 + 10 \equiv 8 \pmod{26},$$

$$Y \equiv 7Q + 10 \equiv 7 \cdot 0 + 10 \equiv 10 \pmod{26},$$

$$Y \equiv 7Q + 10 \equiv 7 \cdot 17 + 10 \equiv 129 \pmod{26},$$

$$Y \equiv 7Q + 10 \equiv 7 \cdot 11 + 10 \equiv 9 \pmod{26},$$

$$Y \equiv 7Q + 10 \equiv 7 \cdot 14 + 10 \equiv 4 \pmod{26},$$

$$Y \equiv 7Q + 10 \equiv 7 \cdot 18 + 10 \equiv 6 \pmod{26},$$

$$Y \equiv 7Q + 10 \equiv 7 \cdot 19 + 10 \equiv 13 \pmod{26}.$$

These calculated results represent the ciphertext (in numbers)

8 10 25 9 4 6 13

or the equivalent letters,

IKZJE GN.

For plaintext recovery, we must solve the following equation:

$$Q \equiv 7^{-1}(Y - 10) \bmod 26.$$

Because 7 is prime to 26, its inverse value is available and it can be observed from Table 2. Consequently, to discover the plaintext from the ciphertext, we can crack it through the following deciphering transformations:

$$Q \equiv 15(Y - 10) \equiv 15 \cdot (8 - 10) \equiv 15 \cdot -2 \equiv 22 \pmod{26},$$

$$Q \equiv 15(Y - 10) \equiv 15 \cdot (10 - 10) \equiv 15 \cdot 0 \equiv 0 \pmod{26},$$

$$Q \equiv 15(Y - 10) \equiv 15 \cdot (25 - 10) \equiv 15 \cdot 15 \equiv 17 \pmod{26},$$

$$Q \equiv 15(Y - 10) \equiv 15 \cdot (9 - 10) \equiv 15 \cdot -1 \equiv 11 \pmod{26},$$

$$Q \equiv 15(Y - 10) \equiv 15 \cdot (4 - 10) \equiv 15 \cdot -6 \equiv 14 \pmod{26},$$

$$Q \equiv 15(Y - 10) \equiv 15 \cdot (6 - 10) \equiv 15 \cdot -4 \equiv 18 \pmod{26},$$

$$Q \equiv 15(Y - 10) \equiv 15 \cdot (13 - 10) \equiv 15 \cdot 3 \equiv 19 \pmod{26}.$$

Accordingly, $Q \equiv 22\ 0\ 17\ 11\ 14\ 18\ 19$, which yields the original plaintext “WAR LOST”. Nevertheless, affine cryptanalysis has the negative aspects of huge computational difficulty and long processing time, especially for long texts.

Table 2. The inverse magnitude of *m*

m	m^{-1}
1	1
3	9
5	4
7	15
9	3
11	19
15	7
17	23
19	11
21	5
23	17
25	25

5. Conclusions

A genetic algorithm for breaking affine ciphers has been used by a certain equation. It has an optimal solution through three operations: selection, crossover and mutation. The gained results to crack the ciphertext of 375 letters show that the fitness percentage value is proportional to number of adopted iterations in the case of affine cryptanalysis using GAs as compared to the same one without using GAs. The optimal results can be seen at 120 generations and the deduction of the full text can be achieved by an English comprehending process. The adoption of GA in affine cryptanalysis improved hugely the fitness value from 65.77% (in the classical affine cryptanalysis) to 95% as well as the remarkable corrected letters. It is obvious from the proposed tool to cryptanalyze affine ciphers that in a short time, it will arrive close enough to the exact plaintext that a visual inspection of the resulting plaintext could be used to conclude any misplaced letters.

Acknowledgement

The authors thank the anonymous referees for their valuable suggestions towards the improvement of the manuscript.

References

[1] A. J. Clark, Genetic optimization heuristics for cryptology, Ph.D. Thesis, Queensland University of Technology, 1998.

[2] Y. S. Mezaal, D. A. Hammood and M. H. Ai, OTP encryption enhancement based on logical operations, 6th International Conference on Digital Information Processing and Communications (ICDIPC), Beirut, 2016.

[3] R. Spillman, M. Janssen, B. Nelson and M. Kepner, Use of a genetic algorithm in the cryptanalysis of simple substitution ciphers, Cryptologia 17(1) (1993), 31-44.

[4] A. J. R. Matthews, The use of genetic algorithm in cryptanalysis, Cryptologia 17(2) (1993), 187-201.

[5] R. Toemeh and S. Arumugam, Applying genetic algorithms for searching key-space of polyalphabetic substitution ciphers, The International Arab Journal of Information Technology 5(1) (2008), 87-91.

[6] R. Turcinhodzic, Graphical presentation of affine transformation cryptanalysis, The 5th International Conference on Computer Science and Education, Hefei, 2010, pp. 1555-1559.

[7] S. S. Omran, A. S. Al-Khali and D. M. Alsaady, Cryptanalytic attack on Vigenère cipher using genetic algorithm, IEEE Conference on Open Systems, 2011, pp. 59-64.

[8] D. M. Alsaady, A comparison between single and multi-crossover points to break Hill cipher using heuristic search: MA & GA, Engineering and Technology Journal 31(4) (2013), 490-504.

[9] S. S. Omran, A. S. Al-Khali and D. M. Alsaady, Using genetic algorithms to cryptanalyse A mono alphabetic cipher by using different mutation rates and lengths of text, First International Scientific Conference of Cihan University, Erbil, 2014, pp. 1-10.

[10] R. S. Habeeb, Arabic text cryptanalysis using genetic algorithm, Iraqi Journal of Electrical and Electronic Engineering 12(2) (2016), 161-166.

[11] A. G. Konheim, Computer Security and Cryptography, Wiley, 2007.

[12] S. N. Sivanandam and S. N. Deepa, Introduction to Genetic Algorithms: Springer, 2008.

[13] R. L. Haupt and S. E. Haupt, Practical Genetic Algorithms, 2nd ed., John Wiley and Sons, 2004.

[14] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, 3rd ed., Springer, 1996.

[15] A. K. Jalal, Direct torque control of induction motor based on intelligent systems, Ph.D. Thesis, University of Technology, Baghdad, Iraq, 2007.