Cryptanalysis using Biological Inspired Computing Approaches

Badrisham bin Ahmad[1] and Mohd Aizaini bin Maarof[2]

Department of Computer Systems and Communications
Faculty of Computer Science and Information Systems
Universiti Teknologi Malaysia
[1]Tel: 07-5532398, E-mail: badrisham@gmail.com
[2]Tel: 07-5530002, E-mail: maarofma@fsksm.utm.my

Abstract

In security, cryptology is introduced to guarantee the safety of data, whereby it is divided into cryptography and cryptanalysis. Cryptography is a technique to conceal information by means of encryption and decryption while cryptanalysis is used to break the encrypted information using some methods. Biological Inspired Computing (BIC) is a method that takes ideas from biology to be used in computing. BIC is a field that has been widely used in many computer applications such as pattern recognition, computer and network security and optimization. Some examples of BIC approaches are genetic algorithm (GA), ant colony and artificial immune system (AIS). GA and ant colony have been successfully applied in cryptanalysis of classical ciphers. Therefore, this paper will review these techniques and explore the potential of using AIS in cryptanalysis.

Keywords: Cryptanalysis, Genetic Algorithm, Artificial Immune System, Ant Colony.

1 Introduction

There are many cryptographic algorithms (cipher) that have been developed for information security purposes such as the Data Encryption Standard (DES), Advanced Encryption Standard (AES) and Rivest-Shamir-Adleman (RSA). These are some examples of a modern cipher. The foundation of the algorithms, especially block ciphers, is mainly based on the concepts of a classical cipher such as substitution and transposition [3]. For instance, DES uses only three simple operator namely substitution, permutation (transposition) and bit-wise exclusive-OR (XOR) [2].

BIC is a field that has caught the interest of many researchers. The ability of using BIC approaches in various fields has been proven. Clark [3] hopes for those who do research in BIC especially related to ants, swarm and artificial immune system, to examine the application of those techniques in cryptology. He also states that a good place to start is on classical cipher cryptanalysis or Boolean function design.

This paper is organized as follows: first, we review simple substitution cipher, columnar transposition cipher and permutation cipher which are types of classical cipher, in Section 2. In Section 3, some biological inspired computing approaches employed are explained and the use of these approaches in cryptanalysis is reviewed in Section 4. Finally, conclusions are given in Section 5.

2 Classical Ciphers

Classical ciphers are often divided into substitution ciphers and transposition ciphers. There are many types of these ciphers. In this paper, we focus on simple substitution cipher and two types of transposition cipher namely columnar transposition cipher and permutation cipher. The ciphers are vulnerable to ciphertext-only attacks by using frequency analysis.

Basically, a simple substitution cipher is a technique of replacing each character with another character. The mapping function of replacing the characters is represented by the key used. For this purpose of study, white spaces are ignored while other special characters like comma and apostrophe are removed. Figure 1 shows the example of a simple substitution cipher:

Alphabet: 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
Key: M N F Q Y A J G R Z K B H S L C I V U D O W T E P X
Example:
Plaintext: MENDAPATKAN SOKONGAN SEBANYAK YANG MUNGKIN
Ciphertext: HYSQMCMDKMS ULKLSJMS UYNMSPMK PMSJ HOSJKRS

Figure 1: Simple substitution cipher

The idea of a transposition cipher is to alter the position of a character to another position. In columnar transposition cipher, the plaintext is written into a table of fixed number of columns. The number of columns depends on the length of the key. The key represents the order of columns that will become the ciphertext. We only consider 26 characters in the alphabet, so all special characters are removed. For example, the plaintext "MENDAPATKAN SOKONGAN SEBANYAK YANG MUNGKIN" with the key "461532" is transformed to ciphertext by inserting it into a table as shown in the example in Figure 2.

4	6	1	5	3	2
M	E	N	D	A	P
A	T	K	A	N	S
O	K	O	N	G	A
N	S	E	B	A	N
Y	A	K	M	U	N
G	K	I	N	J	D

Figure 2: Columnar transposition cipher

Two dummy alphabets (here, J and D) are added for complete the rectangle and the ciphertext can be written in group of five characters [4]. So the ciphertext of this cipher are "NKOEK IPSAN NDANG AUJMA ONYGD ANBMN ETKSA K".

The permutation cipher operates by rearranging each character in a plaintext block by block based on a key. The size of the block is the same as the length of the key and the ciphertext can also be written in group of five characters. Using the same plaintext and key of the previous example, the ciphertext of the permutation cipher is produced as depicted in Figure 3 as follows:

Key: plaintext order: 1 2 3 4 5 6
ciphertext order: 4 6 1 5 3 2

Order: 123456 123456 123456 123456 123456
Plaintext: MENDAP ATKANS OKONGA NSEBAN YAKMUN GKINJD
Ciphertext: DPMAN EASAN KTNAO GOKBN NAEKM NYUKA NDGJI K

Figure 3: Permutation cipher

In both simple substitution and transposition cipher, there are same disadvantage which regards to the frequency of characters. Based on the Figure 1, the character *A* is replaced with *M*, *B* with *N* and so forth. Therefore, the frequency of each character in the plaintext will be exactly the same as the frequency of its corresponding ciphertext character. In the example, the character *B* and its corresponding ciphertext character *N* has a frequency of 17.14%. Hence, the encryption algorithm preserves the frequency of characters of the plaintext in the ciphertext because it merely replaces one character with another. Still, the frequency of characters depends on the length of the text and probably, some characters are not even used in plaintext. As shown in the above example, the character *C*, *R* and *L* are some characters that do not exist in the plaintext. Therefore, many researchers use frequency analysis for cryptanalysis of simple substitution cipher. Analyses were done by using frequency of single character (unigram), double character (bigram), triple character (trigram) and so on (*n*-grams). The technique used to compare candidate keys to the simple substitution cipher is to compare the frequency of *n*-grams of the ciphertext with the language of the text. The general formula used to determine the suitability of a proposed key to a simple substitution cipher is shown in Equation 1 [2].

$$C_k = \alpha g \sum_{i \in A} \left| K^u_{(i)} - D^u_{(i)} \right| + \beta g \sum_{i,j \in A} \left| K^b_{(i,j)} - D^b_{(i,j)} \right| + \gamma g \sum_{i,j,k \in A} \left| K^t_{(i,j,k)} - D^t_{(i,j,k)} \right| \quad (1)$$

In Equation 1, *K* and *D* denote known language statistics and ciphertext statistics respectively. Indices *u*, *b* and *t* refer to unigram, bigram and trigram respectively while α , β and γ allow the assigning of different weight to each *n*-gram. *A* represents the language character such as from A to Z in English language. The formula in Equation 1 also can be used in transposition cipher [1].

In the effort of attacking the transposition cipher, the multiple anagramming attack can be used. The ciphertext is written into a table which the number of columns represents the length of the key. For columnar cipher, the ciphertext is written into the table column by column from left to right while in permutation cipher, the ciphertext is written row by row from top to bottom. After that, the columns are rearranged to form readable plaintext in every row [18].

3 Biological Inspired Computing Approaches

BIC is a method that takes ideas from biology to be used in computing. It relies heavily on the fields of biology, computer science and mathematics. Some of BIC approaches are GA, artificial neural network, AIS, DNA, *Cellular Automata*, *ant colony*, *particle swarm optimization* and *membrane computing*. Three of these techniques namely GA, ant colony and AIS describe later in this section.

3.1 Genetic Algorithm

Genetic Algorithm (GA) is a technique that is used to optimize searching process and was introduced by Holland in 1975 [15]. This algorithm is based on natural selection in the biological sciences [13]. There are several processes in GA namely selection, mating and mutation.

In the beginning of the cycle, a set of random population is created as the first generation. Elements that make up the population are the potential solution to the problem. The population is represented by strings. Then, pairs of strings are selected based on a certain criteria called a fitness function. These pairs are known as parents and will be mated to produce children. The children are then mutated based on a mutation rate because not all children are mutated. After the mutation process, a new set of population is formed (the next generation). The cycle continues until some stopping condition is met such as a maximum number of generations.

This algorithm has been successfully applied in cryptanalysis of classical and modern ciphers such as simple substitution, polyalphabetic, transposition, knapsack, rotor machine, RSA and TEA. We will further explore the usage of this algorithm in cryptanalysis in Section 4.

3.2 Ant Colony Optimization

Ant colony optimization is inspired by the pheromones trail laying and following behavior of real ants which use pheromones as a communication medium. This approach was proposed for solving hard combinatorial optimization problems [11]. An important aspect of ant colonies is the collective action of many ants result in the location of the shortest path between a food source and a nest. Standard ant colony optimization (ACO) algorithm contains probabilistic transition rule, goodness evolution and pheromone updating [6].

In cryptanalysis, ACO algorithm has been applied in breaking transposition cipher and block cipher. Cryptanalysis of transposition cipher published in [1] is reviewed in Section 4 of this paper.

3.3 Artificial Immune System

Artificial immune systems (AIS) can be defined as computational systems inspired by theoretical immunology, observed immune functions, principles and mechanisms in order to solve problems [8]. AIS can be divided to population-based algorithm such as negative selection and clonal selection algorithm and network-based algorithm such as continuous and discrete immune networks. AIS have been applied to a wide variety of application areas such as pattern recognition and classification, optimization, data analysis, computer security and robotic [8]. Hart and Timmis [14] categorized these application areas and some others into three major categories namely learning, anomaly detection and optimization. In optimization, most of the papers published are based on the application of clonal selection principle using the algorithm such as Clonalg, opt-AINET and B-cell algorithm.

De Castro & Von Zuben [5] proposed a computational implementation of the clonal selection algorithm (it is now called Clonalg). The authors compared their algorithm's performance with GA for multi-modal optimization and argue that their algorithm was capable of detecting a high number of sub-optimal solutions, including the global optimum of the function being optimized. Castro and Timmis [7] extended this work by using immune network metaphor for multi-modal optimization.

Clonal selection has also been used in optimization of dynamic functions [21]. The result is compared with evolution strategies (ES) algorithm. The comparison is based on time and performance and shows that clonal selection is better than ES in small dimension problems. However, in higher dimension, ES outperformed the clonal selection in time and performance.

Other than that, Rozi Malim [17] applied the Clonalg in a scheduling problem, with the name clonal selection algorithm for examination timetabling (CSAET). The research shows that CSAET is successful in solving problems related to scheduling. From the comparison performed between CSAET with GA and memetic algorithm, CSAET produced quality output as good as those algorithms.

Therefore, literature shows that AIS is capable of producing good results in various fields especially regarding optimization. It is hoped that AIS will also find its way in cryptanalysis.

4 BIC in cryptanalysis

Classical cipher was successfully attacked using various metaheuristic techniques. Metaheuristic is a heuristic method for solving a very general class of computational problems. Therefore, this technique is commonly used in combinatorial optimization problems. Some of metaheuristic techniques that were successfully applied in the cryptanalysis of classical cipher are genetic algorithm [19], [9], simulated annealing [12], tabu search [2], ant colony optimization [18] and hill climbing [16]. In this paper, we will review BIC techniques (GA and ant colony) that have been successfully applied in cryptanalysis of classical ciphers (simple substitution and transposition cipher).

Spillman et al [19] have published their paper on the cryptanalysis of simple substitution cipher using genetic algorithm in 1993. The paper is an early work done by using GA in cryptanalysis and it is a good choice for re-implementation and comparison [4]. In [19], the authors review some idea about genetic algorithm before they show the steps on how the algorithm is applied in the cryptanalysis of simple substitution cipher. The aim of the attack is to find the possible key values based on frequency of characters in the ciphertext. The key is sorted from the most frequent to the least frequent characters in the English language. In the selection process, pairs of keys (parents) are randomly selected from the population (contains a set of keys that is randomly generated for the first generation) based on fitness function as shown in Equation 2.

$$\left(1-\sum_{i=1}^{26}\left\{\left|SF[i]-DF[i]\right|+\sum_{j=1}^{26}\left|SDF[i,j]-DDF[i,j]\right|\right\}/4\right)^{8} \qquad (2)$$

The fitness function compares unigram and bigram frequencies characters in the known language with the corresponding frequencies in the ciphertext. Keys with higher fitness value have more chance of being selected. Mating is done by combining each of the pairs of parents to produce a pair of children. The children are formed by comparing every element (character) in each pair of parents. After that, one character in the key can be change with a randomly selected character based on a mutation rate in the mutation process. The selection, mating and mutation processes continue until a stopping criterion is met. We have applied the attack using Malay language text and the result is satisfactory based on certain parameters [20].

Another paper published in 1993 utilizing genetic algorithm in cryptanalysis was by Matthews [1]. However, the paper is focuses on transposition cipher. The attack is known as GENALYST. The attack finds the correct key length and correct permutation of the key of a transposition cipher. Matthews uses a list containing ten bigram and trigram yang that have been given weight values to calculate the fitness. For instance, the trigram 'THE' and 'AND' are given a score of '+5' while 'HE' and 'IN' are given a score of '+1'. Matthews also give '-5' score for the trigram of 'EEE'. This is because, although 'E' is very common in English, but a word containing a sequence of three 'E's is very uncommon in normal English text. Higher fitness values have more chance of being selected. After the selection process has been done, mating is performed using a position-based crossover method. Then, the mutation process is applied. There are two possible mutation types that can be applied. First, randomly swap two elements and second, shift forward all elements by a random number of places. The experiment was done by using population size of 20, 25 generations and crossover decreases from 8.0 to 0.5. The result shows that GENALYST is successful in breaking the cipher with key lengths of 7 and 9.

Ant colony optimization has also been successfully implemented in the cryptanalysis of transposition cipher published in [18]. The paper uses specific ant algorithm named Ant Colony System (ACS) with known success on the Traveling Salesman Problem (TSP) [10], to break the cipher. The authors used the *bigram adjacency score*, $Adj_{(I,J)}$ to define the average probability of the bigram created by juxtaposing columns *I* and *J*. The score will be higher for two correctly aligned columns. Other than that, they also used dictionary heuristic, *Dict(M)* for the recognition of plaintext. The authors also made a comparison between the results produced by ACS with the result of previous metaheuristic techniques in transposition cipher which involves differing heuristics, processing time and success criteria. The comparison shows that the ACS algorithm can decrypt cryptograms which are significantly shorter than other methods due to the use of dictionary heuristics in addition to bigrams.

5 Conclusion

This paper reviews works on cryptanalysis of classical ciphers using BIC approaches. The types of classical ciphers involved are the simple substitution and transposition cipher while GA and ant colony optimization is the techniques used. GA has been applied to both ciphers but only transposition cipher was found to have been implemented using ant colony. AIS is also discovered to be a promising approach to be employed in cryptanalysis based on its ability to solve optimization problems. Therefore, the application of AIS in cryptanalysis should be further studied.

6 References

[1] Clark, A. J., (1998) Optimisation Heuristics for Cryptology. *PhD Thesis*, Queensland University of Technology.

[2] Clark, A. J. and Dawson E., (1998) Optimisation Heuristics for the Automated Cryptanalysis of Classical Ciphers. *Journal of Combinatorial Mathematics and Combinatorial Computing*, Papers in honour of Anne Penfold Street, Vol. 28, pp. 63-86.

[3] Clark, J. A. (2003) Invited Paper. Natured-Inspired Cryptography: Past, Present and Future. *IEEE Conference on Evolutionary Computation 2003. Special Session on Evolutionary Computation and Computer Security*. Canberra

[4] Delman, B. (2004) Genetic Algorithm in Cryptography. *Master Thesis* Rochester Institute of Technology, New York.

[5] de Castro, L. N. & Von Zuben, F. J. (2000) The Clonal Selection Algorithm with Engineering Applications. *Proceeding of GECCO'00 –* Workshop Proceedings, pp. 36-37.

[6] de Castro, L. N. (2002) Immune, Swarm and Evolutionary Algorithms Part I: Basic Models. *International Conference on Neural Information Processing* Vol. 3 pp 1464-1468

[7] de Castro, L.N and Timmis, J (2002). An Artificial Immune Network for Multimodal Optimisation. *Proceedings of the Congress on Evolutionary Computation*., Honolulu, Hawaii, USA. pp 699-704

[8] de Castro, L. N., and Timmis, J. (2003). Artificial Immune System as a Novel Soft Computing Paradigm. *Soft Computing 7*. pp. 526-544

[9] Dimovski, A. and Gligoroski, D. (2003) Attacks on Transposition Cipher Using Optimization Heuristics. *Proceedings of ICEST 2003,* Sofia, Bulgaria.

[10] Dorigo, M. and Gambardella, L. M. (1997) Ant Colony System: A Cooporative Learning Approach to the Traveling Salesman Problem. *IEEE Transaction on Evolutionary Computation*, Vol. 1 No. 1 pp 53-66.

[11] Dorigo, M (2000). The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances. *Technical Report*

[12] Giddy, J.P and Safavi-Naini R. (1994) Automated Cryptanalysis of Transposition Ciphers. *The Computer Journal*, Vol. 37 No. 5 pp 429-436.

[13] Goldberg, D., (1989) Genetic Algorithms in Search, Optimization, and Machine Learning. *Reading MA*: Addison-Wesley.

[14] Hart, E. and Timmis, J. (2005) Application Areas of AIS: The Past, the Present and the Future. *ICARIS 2005* pp. 483-497.

[15] Holland, J. H. (1975) Adaptation in Natural and Artificial Systems. Ann Arbor MI: University of Michigan Press.

[16] Jakobsen, T. (1995) A Fast Method for the Cryptanalysis of Substitution Ciphers. *Cryptologia* Vol XIX No. 3 pp. 265-274.

[17] Mohd Rozi Malim, Ahamad Tajudin Khader and Adli Mustafa (2005) A Clonal Selection Algorithm for University Examination Timetabling. *International Symposium on Bio-Inspired Computing*. pp. 1-6.

[18] Russell, M. D., Clark, J. A. and Stepney, S. (2003) Making the Most of Two Heuristics: Breaking Transposition Ciphers with Ants. *CEC 03.*

[19] Spillman, R., Jansen, M., Nelson, B. and Kepner, M. (1993) Use of Genetic Algorithms in Cryptanalysis of Simple Substitution Ciphers. *Cryptologia* Vol. 17 No 1 pp. 31–44.

[20] Subariah Ibrahim, Marina Md Arshad, Muhalim Muhamed Amin, Badrisham Ahmad and Muhammad Reza Z'aba (2005) Cryptanalysis of Substitution Cipher Using Genetic Algorithm. *Simposium Kebangsaan Sains Matematik ke XIII*, Universiti Utara Malaysia.

[21] Walker, J. H. and Garrett, S. M. (2003) Dynamic Function Optimisation: Comparing the Performance of Clonal Selection and Evolutionary Strategies. *ICARIS 2003* pp. 273-284