PARTICLES SWARM OPTIMIZATION FOR THE CRYPTANALYSIS OF TRANSPOSITION CIPHER

Sarab M. Hameed[*] and Dalal N. Hmood[**]
[*]Computer Science Department, College of Science, University of Baghdad.
[**] Computer Science Department, College of science, University of Al-Nahrain.

Abstract

Transposition ciphers are a class of historical encryption algorithms based on rearranging units of plaintext according to some fixed permutation which acts as the secret key. This paper presents a new investigation for cryptanalysis transposition cipher based on Particle Swarm Optimization (PSO). PSO is utilized for the automatic recovery of the key, and hence the plaintext, from only the cipher text. Based upon a mathematical model of the social interactions of swarms, the algorithm has been shown to be effective at finding good solutions. Experimental results show the ability of PSO in finding the correct secret key which is used to recover the plaintext.

Keywords: Evolutionary Algorithms, Particle Swarm Optimization, Transposition Cipher.

1. Introduction

Cryptography has had a long and colorful history. The earliest schemes, now termed the classical ciphers, were designed to be carried out with pen and paper rather than by electronics. Many were transposition cipher [1], also sometimes called a permutation cipher, are one for which applying *encryption to* plaintext produces ciphertext with the same symbols as the plaintext, but rearranged in different positions. Classical cryptography became obsolete after the advent of computers; more complex ciphers could be used, and older ciphers broken with greater ease. Nonetheless, modern analogues of classical schemes can still be found as components of larger ciphers. In particular, some iterated block ciphers, such as the Data Encryption Standard [2], incorporate transpositions to provide diffusion.

The cryptanalyst's tactic when presented with a transposition was to exploit particular statistical features of the ciphertext, as well as to rely upon intuition, luck and trial-and error, to find the correct decryption. As this was sometimes too slow a process, mechanized aids were used as early as World War II [3] by which frequencies of letter pairs (known as *bigrams*) were automatically examined in order to narrow down the space of possible keys. The remaining few keys could then be checked exhaustively by hand to recover the plaintext.

This paper considers the possibility of fully automating this procedure by using Particle Swarm Optimization (PSO). PSO is a stochastic, population-based evolutionary computer algorithm for problem solving. It is a kind of swarm intelligence that is based on social-psychological principles and provides insights into social behavior, as well as contributing to engineering applications. The PSO algorithm was first described in 1995 by James Kennedy and Russell C. Eberhart [4]. The techniques have evolved greatly since then, and the original version of the algorithm is barely recognizable in the current ones.

A simple transposition or permutation cipher works by breaking a message into fixed size blocks, and then permuting the characters within each block according to a fixed permutation, say *P*. The key to the transposition cipher is simply the permutation *P*. So, the transposition cipher has the property that the encrypted message contains all the characters that were in the plaintext message. Let's consider an example of a transposition cipher with a period of three and a key *P*= {3, 1, 2}. In this case, the message is broken into blocks of three characters, and after encryption the third character in the block will be moved to position one, the first character in the block will be moved to position two, and the second character in the block will be moved to position three. The decryption can be achieved by following the same process as encryption using the inverse of the encryption permutation. In this case the decryption key, P^{-1} is equal to {2, 3, 1}.

Key: 312
Plaintext : CRYPTOGRAPHY
Ciphertext: YCROPTAGRYPH

Breaking transposition has been carried out by many researchers. Methew [6] presented an attack on transposition cipher using genetic algorithm. Giddy and Safavi-Naini [7] have published an attack on the transposition cipher using simulated annealing. Russell [8] presented an attack on transposition cipher using ant colony. R. Garg P [9] proposed a cryptanalysis method based on genetic algorithm, tabu search & simulated annealing to break a transposition cipher. Toemeh [10] presented an attack on transposition cipher using genetic algorithm. Werner R. Grundlingh [11] presented an attack on the simple cryptographic cipher using genetic algorithm.

The objectives of the paper are:

- To determine the ability of PSO on the cryptanalysis of transposition cipher.
- To determine the effect of initial entry parameter on PSO

The rest of this paper is organized as follows. Section two presents description of cryptanalysis transposition cipher using PSO. Section three explains some results and discussions. Finally, Section four introduces conclusion.

2. PSO Attack on Transposition Cipher

This section states how the PSO was used to break transposition cipher (i.e. finding the correct key that is used to encrypt the plaintext).The particle swarm optimization method is an algorithm based on machine learning processes that is used for cracking transposition cipher.

The algorithm starts by choosing a random population of potential solutions, each of which is called a particle. Each particle keeps the coordinates of its best position found so far, called *pbest*. The best position of all particles is held in *gbest*. The particles change coordinates towards *gbest* according to the velocity updated. If the new coordinates of the particle produce a better result than the current *pbest*, then the *pbest* is updated accordingly and compared to *gbest* for possible update [12].

Fig. (1) depicts the process of encryption and decryption in the transposition cipher and the cryptanalysis with PSO.

2.1 Particle Representation and Initial Swarm

For cracking transposition cipher, the coordinates of the position of a particle (x_{id}) is an integer number referred to permutation of the key. A population of particles is constructed randomly for the PSO algorithm for attacking transposition cipher. For each particle, an integer value assigned randomly. Here the dimension of the particle is one.

For example, suppose the key length is four, then the particle value assigned randomly in range [0,23]. This range value converts to an integer permutation using simple function that convert the integer number to integer permutation. Table (1) illustrates the mapping between integers and integer permutations.

Additionally, Velocity values are restricted to some minimum and maximum values using piece-wise function [$V_{max,}$ V_{min}] where V_{min}= -V_{max} , the velocity of particle *i* is given in equation 1.

$$v_{id}^{\ o} = V_{\min} + [V_{\max} - V_{\min}] * rand() \quad \text{.....(1)}$$

Where rand() are random function in the range [0,1]. This limit enhances the local search exploration of the problem space. The V_{max} is set to 2.

Table (1)
Mapping between Integers and Integer Permutations.

Permutation	*Particle i*
0321	0
0312	1
0132	2
0123	3
0231	4
0213	5
1023	6
1032	7
1302	8
1320	9
1203	10
1230	11
2031	12
2013	13
2103	14
2130	15
2301	16
2310	17
3021	18
3012	19
3102	20
3120	21
3201	22
3210	23

2.2 Particle Evaluation

x_{id} is converted to an integer permutation using simple function to specify the that is used to decrypt the ciphertext. The technique used to compare candidate key is to compare n-gram (i.e. a subsequence of N-items in any sequence) statistics of the decrypted message with those of the language (which are assumed known). Equation 1 is a formula used to determine the suitability of a proposed key [13].

$$Fitness = \alpha \sum_{i,j \in A} \left| K^{b}_{i,j} - D^{b}_{i,j} \right| + \beta \sum_{i,j,k \in A} \left| K^{t}_{i,j,k} - D^{t}_{i,j,k} \right| \quad \text{.....(2)}$$

Where

K is known as language statistics. In this paper English language is used.

A denotes language alphabet (i.e. for English language A…Z).

D is the decrypted message statistics

b and *t* are the bigram (2-gram)and trigram (3-gram) statistics.

The values of α and β allow assigning of different weights to each of the two n-gram types (i.e. bigram and trigram). The range of α and β are [0, 1].

In the process of determining the cost associated with a transposition cipher key the proposed key (i.e. evaluate particle *i*) is used to decrypt the ciphertext and then the statistics of the decrypted message are then compared with statistics of the language.

2.3 Finding New Solutions

The new solutions are found by updating the velocity and dimension respectively.

First, updating the velocity as formulated in equation (3) [14].

$$v^{k}_{id} = v^{k-1}_{id} + c_1 rand()(pb^{k-1}_{id} - x^{k-1}_{id}) + c_2 Rand()(gb^{k-1}_{id} - x^{k-1}_{id}) \quad \text{.....(3)}$$

Where rand() and Rand() are uniformly distributed random variables in the range[0,1]and c_1, and c_2 are learning factors. In this research, we set them to 2.

Then, updating the dimension *d* of the particle *i* as formulated in equation 4.

$$x^{k}_{id} = x^{k-1}_{id} + v^{k-1}_{id} \quad \text{.....(4)}$$

3. Results and Discussions

All experiments presented in this paper were performed on text using capital English characters alphabet, i.e. A-Z. All punctuation and structure (sentences/paragraphs) has been removed from the text before encryption. The algorithm has been implemented successfully on different amount of ciphertext provided to attack. The algorithm runs with different ciphertext, key size, and swarm size.

Experimental results show that particle swarm optimization is automatic tool for breaking transposition ciphers as long as bigram and trigram is used to calculate the fitness of particles. Here for each size there are some keys have been broken fully. If the ciphertext and swarm size are having more size the breakable key is more.

Table (2) illustrates the results of of PSO attack on transposition cipher with different

amount of ciphertext swarm size and different key size.

Table (3) shows the required time to break the ciphertext with 250 letters and keys of different size.

Table (2)
The Amount of Keys Recovered Versus Available Ciphertext.

Swarm Size	*Amount of cipher length (letters)*	*Key size*					
		Recovered key					
		5	*6*	*7*	*8*	*9*	*10*
5	50	5	6	7	8	7	7
	100	5	6	7	8	8	8
	150	5	6	7	8	9	9
	200	5	6	7	8	9	10
	250	5	6	7	8	9	10
8	50	5	6	7	8	7	8
	100	5	6	7	8	8	9
	150	5	6	7	8	9	9
	200	5	6	7	8	9	10
	250	5	6	7	8	9	10
10	50	5	6	7	8	7	8
	100	5	6	7	8	8	9
	150	5	6	7	8	9	9
	200	5	6	7	8	9	10
	250	5	6	7	8	9	10
12	50	5	6	7	8	9	10
	100	5	6	7	8	9	10
	150	5	6	7	8	9	10
	200	5	6	7	8	9	10
	250	5	6	7	8	9	10
15	50	5	6	7	8	9	10
	100	5	6	7	8	9	10
	150	5	6	7	8	9	10
	200	5	6	7	8	9	10
	250	5	6	7	8	9	10

Table (3)
Required Times to Break the Ciphertext with 250 Letters for Different Keys Size and Swarm Size 15.

Key size	*5*	*6*	*7*	*8*	*9*	*10*
Time (sec)	0.5	0.5	0.54	0.55	0.55	1.3

4. Conclusions

The paper explains PSO attack on the transposition cipher. PSO was used to generate keys and choice the correct key (secret key) to break the ciphertext and to recover the plaintext. Experimental results show that the PSO is efficient in determining the optimal choice of key to find the plaintext. The number of possible visited keys required for breaking the key (i.e. recover plaintext) is less than the number of possible visited keys required in the Brute Force attack, because the number of possible visited keys possible keys for a transposition cipher with *N* key size is *N* factorial.

References

[1] W. Mao, “Modern Cryptography: Theory & Practice”, pper Saddle River, NJ: Prentice Hall PTR, 2004.

[2] F. L Bauer, "Decrypted Secrets", Springer, second edition, 1997.

[3] National Institute of Standards and Technology (NIST), "Data Encryption Standard (DES)", Federal Information Processing Standards Publication (FIPS PUB) 46-3, October 1999.

[4] C. Russel Eberhart and J. Kennedy, "A New Optimizer Using Particle Swarm Theory", In Proceedings of the Sixth International Symposium on Micro Machine and Human Science MHS ’95, pages 39–43. IEEE Press, October 1995.

[5] J. Kennedy and C. Russel Eberhart, "Particle Swarm Optimization", In Proceedings of IEEE International Conference on Neural Networks, Vol. 4, Perth, WA, Australia, 1995, pp 1942–1948.

[6] R.A.J .Methew, "The Use of Genetic Algorithms in Cryptanalysis", Cryptologia, Vol. 7, No. 4 April 1993, pp 187-201.
[7] J. P Giddy and R Safavi-Naini., "Automated Cryptanalysis of Transposition Ciphers", the Computer Journal, Vol. 37, No. 5, 1994.
[8] M.D. Russell, J.A Clark, and S Stepney, "Making the Most of Two Heuristics: Breaking Transposition Ciphers with Ants Evolutionary Computation", CEC 03, Vol.4, Dec. 2003, pp2653 - 2658.
[9] P. Garg, "Genetic Algorithms, Tabu Search and Simulated Annealing: A Comparison between Three Approaches for the Cryptanalysis of Transposition Cipher", Journal of Theoretical and Applied Information Technology, 2005.
[10] R. Toemeh and S. Arumugam, "Breaking Transposition Cipher with Genetic Algorithm", Electronics and Electrical Engineering, No. 7(79), 2007, pp 75–78.
[11] R.G. Werner and J. H. Van Vuuren, "Using Genetic Algorithm to Break a Simple Cryptographic Cipher", Article, http://www.apprendreen- ligne.net/ crypto/ bibliotheque
[12] M. F.Uddin and A. M.Youssef, "Cryptanalysis of simple substitution ciphers using particleswarm optimization", IEEE Congress on Evolutionary Computation, Vancouver, BC, Canada, July 16-21, 2006.
[13] A. Dimovski, D. Gligoroski , "Attacks on the Transposition Ciphers Using Optimization Heuristics", International Scientific Conference on Information, Communication & Energy Systems & Technologies ICEST 2003, Sofia, Bulgaria, October 2003.
[14] S. Yuhui "Particle Swarm Optimization", IEEE Neural networks Society, 2004.

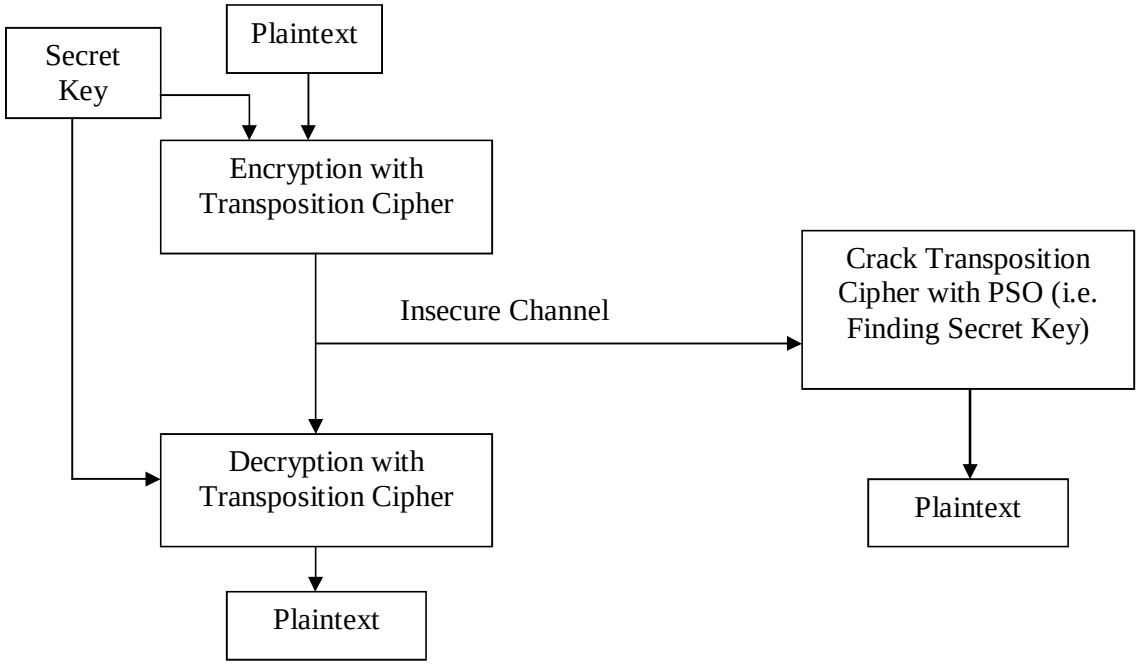

Fig. (1) : Transposition Cipher and Cryptanalysis by Means of PSO.

الخلاصة

شفرة إبدال الموضع هي صنف من خوارزميات التشفيرِ التأريخيةِ المعتمدة على اعادة ترتيب احرف النص الصريح طبقاً لبَعْض التقليبِ الثابتِ الذي يَفْعلُ كالمفتاحِ السريِ. يقدم البحث دراسة جديدة لتحليل شفرة إبدال الموضع بالاعتماد على أمثلية سرب الطيور(PSO). استفيد من PSO لاكتشاف المفتاح بطريقة آلية ومن ثم استخراج النص الصريح من النص المشفر فقط. بالاعتماد على نموذج رياضي من التفاعلات الاجتماعية لسرب، تبين بان الخوارزمية فعال في ايجاد الحلول الجيدة. اثبت نتائج التجارب بقابلية PSO في ايجاد المفتاح السري الصحيح الذي يستخدم لاسترجاع النص الصريح.