Deciphering of Transposition Ciphers using Genetic Algorithm

[1] Alok Singh Jadaun, [2] Vikas Chaudhary, [3] Lavkush Sharma, [4] Gajendra Pal Singh

[1, 2] Department Of Computer Science & Engineering
Bhagwant University Ajmer, Rajasthan-305023, India

[3, 4] Department of Computer Science & Engineering
Raja Balwant Singh Engineering Technical Campus
Bichpuri, Agra, U.P-283105, India

Abstract— Using evolutionary computing for cryptanalysis transposition cipher is an effective and time saving technique. Genetic algorithm also comes under evolutionary computing. These are the simple random section procedure. Transposition cipher is the permutation of the plain text in which only the places of the letters are change according to a particular key which is a group of integer. Cryptanalysis of transposition cipher is a process of finding the key used by the encryption algorithm which can be effectively done using Genetic Algorithm. This paper presents the application of Genetic Algorithm.

Keywords- *Cryptanalyze, Genetic Algorithms, Transposition Cipher.*

1. Introduction

CRYPTANALYSIS is that part of cryptology which deals with the breaking of cipher text without having the knowledge of various enciphering parameters. Cryptology is subdivided into cryptography and cryptanalysis. Cryptography is concerned with the design of cryptosystems, while cryptanalysis is the studying of those methods and techniques which breaks cryptosystems. Both these aspects are closely related and plays vital role in setting up a new cryptosystem and also in the analysis of an existing system. The main focus of this paper is transposition ciphers which are mainly the permutation of places of the letter of plaintext. We are using the bigram analysis of the cipher text. GA are used as a tool for finding the permutation or key used in encryption process so as to generate back the original sequence of the letters. There are numerous papers published which elaborate the application of genetic algorithm in cryptanalysis of various kinds of cipher. Matthews [1] in 1993 presented the idea of using genetic algorithm in cryptanalysis of transposition cipher. He used the following fitness as in Equation 1.

$$F_L = L \sum^{Q} \frac{(P_i S_i)}{100} \qquad (1)$$

R. Toemeh and S. Arumugam in 2007 also presented their work on the cryptanalysis of transposition cipher using genetic algorithm.

$$C_K = \beta \sum_{i,j \epsilon A} |K^b_{(i,j)} - D^b_{(i,j)}| + \gamma \sum_{i,j \epsilon A} |K^t_{(i,j,k)} - D^t_{(i,j,k)}| \quad (2)$$

Equation 2 is a general formula used to determine the suitability of a proposed key (k). Here, A denotes the language alphabet (i.e., for English, [A... Z, _], where _ represents the space symbol), K and D denote known language statistics and decrypted message statistics, respectively, and the indices b and t denote the bigram and trigram statistics, respectively. The values of β and γ allow assigning of different weights to each of the two n-gram types.

2. Transposition Cipher

A transposition is not a permutation of alphabet characters, but a permutation of places [1].
A transposition ciphering algorithms breaks whole text into fixed size block. The greater is the block length the more difficult is its cryptanalysis as the length of text to be decrypted increases. Then the places of letters of each block are interchanged according to a fixed permutation, say P.

$$P = \{x | x \leq L \text{ and } x \in N \qquad (3)$$
$$n(P) = L \qquad (4)$$

Equation 3 and Equation 4 says P is a set of length L consisting natural numbers from 1 to L.
As there is no addition or removal of letters, all the words present in the plaintext are also present in ciphertext. In this method, the message is written in a rectangle, row by row as shown in Fig. 1. Reading the message off, row by row, but permuting the order of the columns. The order of the columns then becomes the key to the algorithm [2, 3, 8]. For example, The plain text is: hello how are you I am fine bye.
The cipher text is:

LWOLwantAHHEROOFMUIAEYNIYXXEXXEBXX
And the key is: P ={3,8,7,4,9,6,1,2,10,5}

Plain Text									
1	2	3	4	5	6	7	8	9	10
H	E	L	L	O	H	O	W	A	R
E	Y	O	U	I	A	M	F	I	N
E	B	Y	E	X	X	X	X	X	X

Cipher Text									
3	8	7	4	9	6	1	2	10	5
L	W	O	L	A	H	H	E	R	O
O	F	M	U	I	A	E	Y	N	I
Y	X	X	E	X	X	E	B	X	X

Fig. 1: Example of Transposition Ciphering Algorithm

In this case, the message is broken into blocks of ten characters, and if there is any blank space left in the last block then consecutive „X" are added to fill all the blanks. After encryption the third character in the block will be moved to position 1, the eight character in the block will be moved to position 2, the seventh is moved to position 3, the fourth to position 4, the ninth to position 5, the sixth to position 6, the first to the position 7, the second to the position 8, the tenth to the position 9 and the fifth to position 10.

3. Cryptanalysis

It is the process of getting the true information from the encrypted text. There is more than one type of attack for cryptanalysis. Which attack to be used depends on the type of ciphertext to be decrypted and the available computational resources? However there are some methods which can be used on every kind of ciphers but they use maximum computational power. For example, Brute Force attack the attacker tries every possible key on a piece of ciphertext until an intelligible translation into plaintext is obtained [4].

A typical cipher takes a clear text message (known as the plaintext) and some secret keying data (known as the key) as its input and produces a scrambled (or encrypted) version of the original message (known as the ciphertext). An attack on a cipher can make use of the ciphertext alone or it can make use of some plaintext and its corresponding ciphertext (referred to as a known plaintext attack) [5].

4. Genetic Algorithm

The genetic algorithm is based upon Darwinian evolution theory [6]. Genetic algorithms have been developed by John Holland, his colleagues, and his students at the University of Michigan.

A. How are GAs different from traditional methods?
In order to surpass their more traditional cousins in the quest for robustness, GAs must differ in some very fundamental ways. Genetic Algorithms are different from more normal optimization and search procedures in four ways [7]:

1. GAs work with a coding of the parameter set, not the parameters themselves.
2. GAs search from a population of points, not a single point.
3. GAs use payoff (objective function) information, not derivatives or other auxiliary knowledge.
4. GAs use probabilistic transition rules, not deterministic rules.

B. Fundamentals of GAs

GA encodes the decision variables of a search problem into finite-length strings of alphabets of certain cardinality. The strings which are candidate solutions to the search problem are referred to as chromosomes, the alphabets are referred to as genes and the values of genes are called alleles. For example, in a problem such as the traveling salesman problem, a chromosome represents a route, and a gene may represent a city. In contrast to traditional optimization techniques, GAs work with coding of parameters, rather than the parameters themselves.

GA are good at taking large, potentially huge search spaces and navigating them, looking for optimal combinations of things, the solutions one might not otherwise find in a lifetime.

GA is simply a cycle of the following genetic operations; graphically it can be shown by Fig. 2 [8].

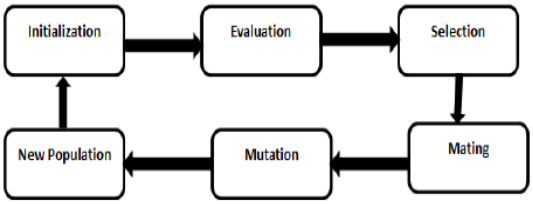

Fig. 2: Cycle of a typical Genetic Algorithm

- Initialization
In this phase new generation is initialized. Basically it is the process of generating a random numbers of candidate solutions.

- Evaluation
This is the process of deciding who shall live and who shall die. Since GA works over a set of random solutions it is important to judge which solution should be considered for the further processing? This part is about calculating the fitness measure of the solutions generated during the execution of GA.

- Selection
Selection is the stage of a genetic algorithm in which individual genomes are chosen from a population for

later breeding (recombination or crossover). Selection is a method that randomly picks chromosomes out of the population according to their evaluation function. The higher the fitness function, the more chance an individual has to get selected. The selection pressure is defined as the degree to which the better individuals are favored. The higher the selection pressured, the more the better individuals are favored. This selection pressure drives the GA to improve the population fitness over the successive generations.

- Mating or Crossover

Crossover is a process of taking more than one parent solutions and producing a child solution from them. After the selection process, the population is enriched with better individuals. The child produced have properties of both the parents.

- Mutation

Mutation alters one or more gene values in a chromosome from its initial state. Mutation is used for randomly altering a apart of an individual to produce a new individual.

- New population

It is the set of those individual solutions which are through the above genetic operators.

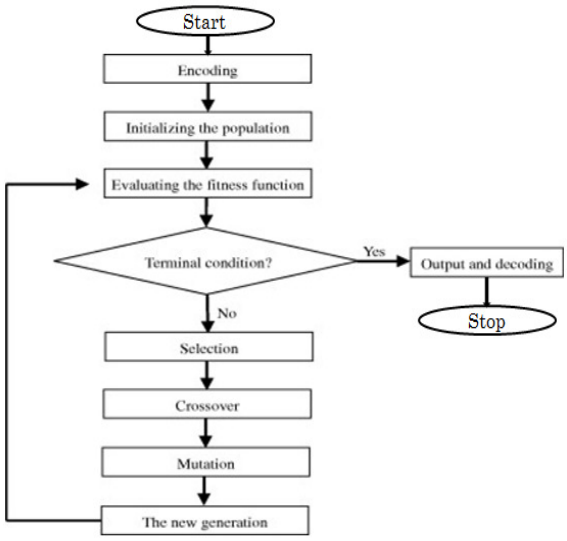

Fig. 3: Flowchart of a GA

5. GA in Cryptanalysis

Prior to using the GA for deriving a solution, the problem must be represented in a manner such that genetic operations can be performed on the individual candid solutions. We have used string representation i.e. set of integers as shown in Fig. 4.

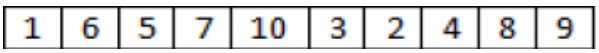

Fig. 4: Set of Integer

After applying the mutation operation we get the solution shown in Fig. 5

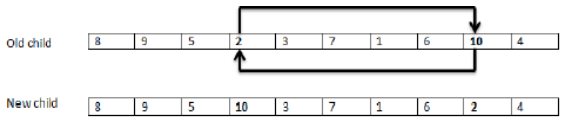

Fig. 5: Mutation of an Individual Solution

Crossover operator manipulates the individual solution as shown in Fig. 6

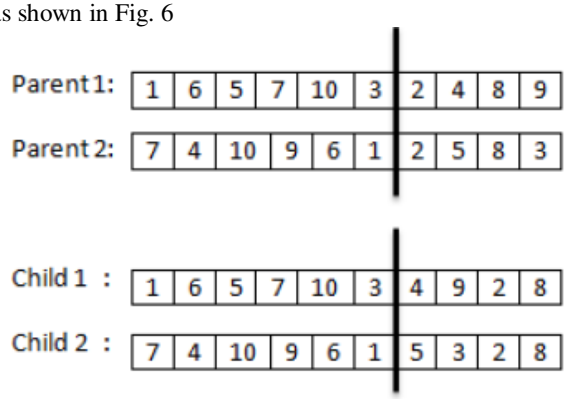

Fig. 6: Crossover of two parent solutions

The Fig. 7 shows the flow chart of the approach used for crypt analyzing transposition cipher using GA.

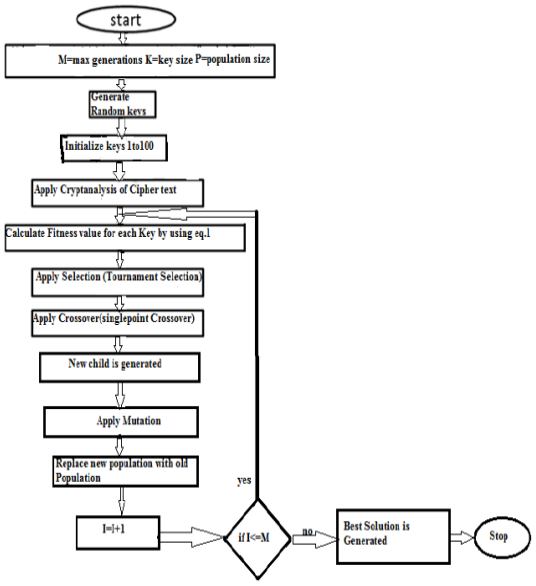

Fig. 7: Flowchart of a typical GA

6. Who Should Live Who Should Die

Genetic algorithm needs some means to decide which member of the population should die and which one should be carried forward so that genetic diversity is maintained. For this purpose we have used a mathematical Equation 3.

$$F(key) = 1 - [\beta \sum_{i,j \in A} |K^b_{(i,j)} - D^b_{(i,j)}|] \quad (3)$$

Where F(key) is fitness value to find optimal solution, $K^b_{(i,j)}$ are the known language bi-gram statistics as shown in table 1, $D^b_{(i,j)}$ are the bigram statistics of the message decrypted with key K. The weight β can be varied to allow more or less emphasis on particular statistics.

7. Results

The Figure 8 shows the relationship between the population size and fitness values %. It is calculated for 400 generations. It is clear from the table that the best fitness value was 50.074 for the number of population 60. Table shows the fitness values for population size (10, 20, 30, 40, 50,60,70,80 and 90).

Table 2. Fitness values for various population size.

Population size	Fitness value %
10	30.216
20	45.052
30	42.255
40	42.07
50	33.216
60	50.074
70	45.254
80	48.21
90	36.410

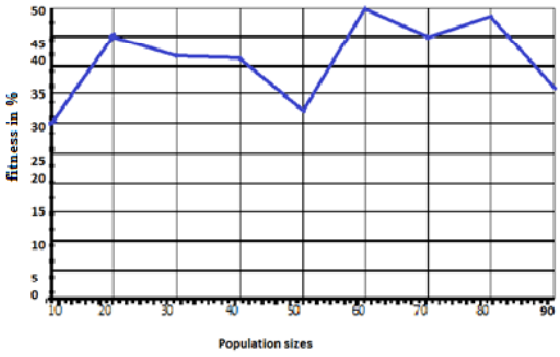

Fig. 8: Graph between population size and fitness values (in %)

8. Conclusion

It was noticed that with increasing key length there was a rapid decrease in the success rate i.e. after the key length value reached up to 6 the success rate was once in seven attempts with 20 numbers of generations. Hence one should increase the maximum number of generation as the key length is increased.

References

[1] Clark, A.J., "*Optimization Heuristics for Cryptology*". Thesis PhD, in Information Security Research Centre, Faculty of Information Technology, Queensland University of Technology, February 1998.

[2] Stallings, W., "*Cryptography And Network Security, Principle And Practices*", 3rd Edition, Pearson Education, 2005.

[3] Rolf,O., "*Contemporary Cryptography*" Artech House Computer Security Series, Boston- London, 2005.

[4] William Stallings., *"Cryptography and Network Security, Principles and Practices"*, 3rd edition, Pearson Education, 2004.

[5] Andrew John Clark, *"Optimization Heuristics for Cryptology"* / PhD thesis, Information Security Research Centre Faculty of Information Technology Queensland University of Technology, February 1998.

[6] Andrew John Clark, *"Optimization Heuristics for Cryptology"* / PhD thesis, Information Security Research Centre Faculty of Information Technology Queensland University of Technology, February 1998.

[7] David E. Goldberg, "Genetic Algorithms in search, Optimization and Machine Learning,", New Delhi: Dorling Kindersley (India), pp. 7.

[8] A.S.Al-Khalid, S.S. Omran Dalal, A.Hammood, "*Using Genetic Algorithms To Break A Simple Transposition Cipher",* Foundation of Technical Education College Of Elec. & Electronic Techniques (Baghdad-IRAQ), May 8, 2013.

[9] R.Toemeh, Department of computer Science and Engineering, Government College of Technology, S.Arumugam, Additional Director, Directorate of technical education, Chennai, India, *"Breaking Transposition Cipher With Genetic Algorithm"*,2007.

[10] Karel P.Bergmann, Renate Scheidler, Christian Jacob, *"Cryptanalysis Using Genetic Algorithms",* University Of Calagary 2500 University Dr. NW Calgary, AB, Canda, Page no 1099,1100.

Alok Singh Jadaun is currently persuing his M.Tech (C.S.E. II Year) from Bhagwant University Ajmer (Rajasthan).

Er. Vikas Chaudhary is currently working in Department of Computer Science & Engineering as a Head of Department in Bhagwant University Ajmer. He is M.tech(cs).

Er. Lavkush Sharma is currently working in Department in Computer Science & Engineering as a Assistant Professor in Raja Balwant Singh Engineering Technical Campus Bichpuri Agra. He is M.tech(C.S.) and 10 years teaching experience.

Gajendra Pal Singh is currently persuing B.Tech (IV) year in Computer Science & Engineering from Raja Balwant Singh Engineering & Technical Campus Bichpuri Agra.

Table 1: Dipthong frequencies for English language

	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
A	6	18	24	26	9	8	13	41	39	0	15	79	36	193	1	12	0	68	61	83	11	29	11	0	34	3
B	12	2	0	0	57	0	0	0	6	0	0	11	0	0	15	0	0	16	2	1	21	0	0	0	10	0
C	27	0	4	0	29	0	0	36	8	0	8	6	0	0	32	0	0	6	0	5	6	0	0	0	1	0
D	56	15	7	8	44	12	8	22	42	3	2	7	11	10	45	5	0	15	31	65	5	1	20	0	13	1
E	105	24	41	92	36	37	18	61	46	5	12	61	55	102	40	32	1	149	135	84	5	25	47	3	43	2
F	24	3	2	2	18	10	5	8	26	5	1	6	6	1	56	2	0	18	5	47	5	0	2	0	11	1
G	22	2	1	2	16	2	1	26	12	0	1	4	2	3	34	1	0	10	11	14	3	0	3	0	5	1
H	135	5	3	2	373	4	2	10	105	1	1	2	4	2	72	2	0	10	9	37	9	0	30	0	10	0
I	15	4	24	34	23	15	18	5	0	1	5	52	33	151	16	5	1	27	92	81	0	18	8	1	0	1
J	4	0	0	0	11	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	6	0	0	0	0	0
K	4	1	0	0	27	1	0	1	19	0	0	1	1	6	3	0	0	0	4	4	0	0	1	0	1	0
L	35	12	4	30	58	8	3	5	35	1	3	88	5	6	32	3	0	2	18	26	2	5	5	0	17	0
M	52	10	1	1	59	3	1	4	20	1	0	1	7	2	27	6	0	1	8	15	5	0	6	0	20	0
N	37	6	16	160	54	6	72	15	23	3	4	4	6	5	56	3	0	2	35	105	4	1	10	0	14	0
O	14	10	7	30	6	121	6	15	12	2	11	22	48	92	19	16	0	98	27	54	139	13	35	1	12	1
P	13	1	0	0	25	1	0	11	7	0	0	17	1	0	12	6	0	19	3	10	5	0	1	0	1	0
Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0
R	47	7	7	28	126	9	8	13	54	1	6	6	10	16	57	5	0	6	39	45	12	7	9	0	18	0
S	83	12	12	7	84	13	5	68	35	1	4	7	8	10	64	18	0	12	41	88	13	1	24	0	10	0
T	41	9	5	6	54	8	3	416	49	2	2	7	7	4	110	3	0	21	29	41	12	0	20	0	22	0
U	11	6	6	10	4	2	11	5	8	0	1	17	6	24	1	11	0	51	47	43	0	0	4	0	1	0
V	7	0	0	0	75	0	0	0	15	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0
W	35	1	0	1	54	1	0	55	55	0	0	2	1	9	21	0	0	3	4	6	0	0	2	0	1	0
X	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Y	35	6	4	5	12	6	3	8	10	0	1	4	6	3	85	6	0	3	21	17	1	0	9	0	3	0
Z	2	0	0	0	3	0	0	0	2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1