Journal of Computer Sciences and Applications, 2015, Vol. 3, No. 2, 52-55
Available online at http://pubs.sciepub.com/jcsa/3/2/6

DOI:10.12691/jcsa-3-2-6

Cryptanalysis Using Soft Computing Techniques

Harsh Bhasin*, Asif Hameed Khan

Department of Computer Sc.&Engg, Jamia Hamdard University, New Delhi-62, India
*Corresponding author: i_harsh_bhasin@yahoo.com

Received March 18, 2015; Revised April 03, 2015; Accepted April 16, 2015

Abstract This paper proposes a Genetic Algorithm (GAs) based cryptanalysis technique. Genetic Algorithms are the optimization techniques which are also known for robustness. The analysis and the involved theory have been presented in the paper. The designing of fitness function has been done using the statistical analysis of a Standard English Language documents. The technique has been verified using 9 text documents of about 4000 words and the results are encouraging. The technique paves way of Soft Computing techniques in Cryptanalysis.

Keywords: *Cryptanalysis, Cryptography, Genetic Algorithms (GAs)*

Cite This Article: Harsh Bhasin, and Asif Hameed Khan, "Cryptanalysis Using Soft Computing Techniques." *Journal of Computer Sciences and Applications*, vol. 3, no. 2 (2015): 52-55. doi: 10.12691/jcsa-3-2-6.

1. Introduction

Cryptanalysis is a procedure of transforming a cipher text into a plaintext. It can also be defined as the study of ciphers, cipher text and cryptosystems [1]. The procedure calls for the retrieval of the plaintext from the cipher text without knowing the cryptographic key or algorithm [1,25]. The traditional cryptanalysis techniques like mono-alphabetic and poly-alphabetic Substitution cipher, Permutation cipher, Transposition cipher, Merkle-Hellman Knapsack cipher, Chor-Rivest Knapsack cipher and Vernam cipher have commonly been used in order to decipher the given cipher text [2]. However, a robust model of accomplishing this task using Genetic Algorithms still eludes the fraternity.

GAs are heuristic search processes based on the concept of survival of the fittest [2]. The algorithms are generally used for optimization problems. Their ability to find the solution using standard operators is remarkable. The algorithms have been successfully used in problems like Travelling Salesman Problem, N- Puzzle etcetera [3-9].

GAs require of setting various algorithm components and parameters for their efficient performance and efficacy [10,11,12,13]. The operators used in GAs are crossover, mutation, selection amongst many others. These operators have been defined in the following sections. The crafting of an appropriate fitness function is the crux of their success. The present work explores the applicability of GAs in the intricate problem of cryptanalysis.

The rest of the paper has been organized as follows. Section 2 presents a brief review of GAs. Section 3 gives a brief overview of related work done on cryptanalysis. Section 4 discusses the Literature review. Section 5 presents the proposed algorithm and section 6 presents the analysis. The last section concludes. The work is sure to pave way of GAs in cryptanalysis.

2. Genetic Algorithm

GAs are inspired by Darwin's theory of evolution [2,14]. According to this theory the best chromosomes, that is those ones having higher values of fitness functions, should survive and create new offspring. GAs constitutes evolutionary computing, which is a rapidly growing area of Artificial Intelligence [2,15]. GAs are heuristic in nature and known for robustness. It gives useful solutions to optimization and search problems [2].

GAs is starts with a set of solutions called population which are represented by chromosomes. The solution from one population are taken and used to form a new population with better fitness. Population is nothing but chromosomes which are generally binary. The population can be refined by using following operators.

Crossover – It is a process of creating a offspring by copying attributes from parent's chromosomes. Number of crossover depends on crossover rate which is generally 2 to 5%. The formula for the number of crossovers is as follows.

Number of crossover=(No. of cells in chromosomes*No. of chromosomes*crossover rate)/200

There are many types of crossovers. Some of them are as follows.

1) *Singe point crossover*- In this type, one crossover point is selected. Binary string from the beginning of chromosome to the crossover point is copied from one parent; the rest is copied from the second parent [1].
2) *Two point crossover*- In this type, the two crossover points are selected and the crossover is implemented by taking some part from the first chromosome, some from the second and the rest from the first [1].
3) *Uniform crossover*- In this type, the bits are randomly copied from the first or from the second parent [1].

The present work uses single point crossover.

2.1. Mutation

Mutation is a process that is carried out in order to break the local maxima. The process is implemented by flipping a random bit from a random chromosome. The number of mutations is calculated using the following formula. Here, the mutation rate is generally very low. The formula for the number of mutations is as follows.

Number of Mutation=(No. of cells in a chromosomes*No. of chromosomes*mutation rate)/200

2.2. Selection

Selection is a process in which individual chromosomes are chosen so as to form a population for crossover [2]. The chromosomes having higher fitness value will be considered better. There are many methods of performing selection, some of them are Roulette-wheel selection, Stochastic universal selection, Tournament selection, Truncation selection etcetera.

2.3. Population size and Fitness function

Deciding the appropriate size of population has been one of the most contentious points [16,17,18]. The population size should neither nor too small nor too large, Too small size of the population could guide the algorithm to poor solutions [18,19,20]. The numerical experiments show that increasing the size of the population of 5 to 100 significantly improves the resulting value. Taking a large population initially considerably increases the running time.

A fitness function is a type of objective function that is used to summarize, how close a given design solution is achieving the set aims [16]. The designing of an apt fitness function guarantees early convergence. As a matter of fact the designing of an appropriate fitness functions is the most important task in the problem reduction step of the problem.

3. Related Work

Cryptanalysis has been of the most researched topics. The research on this topic has been on rise since its inception. Many techniques have been proposed in order to handle the problem of cryptanalysis. However, the main focus has been on classical ciphers, including substitution, permutation, transposition, knapsack and vernam ciphers.

One of the first papers published was that Spillman, Janssen, Nelson and Kepner in 1993 [23]. The work focused on the cryptanalysis of a simple substitution cipher using a GAs. Another paper published in 1993 by R.A. J. Matthews [24], uses an order-based genetic algorithm to attack a simple transposition cipher. Work by R. Spillman [25], applies a genetic algorithm approach to a Merkle-Hellman knapsack system. It may be stated here that the year-1994 saw a few major papers on the topic one of which was by Andrew Clark [26], includes GAs as one of three optimization algorithms applied to cryptanalysis. In 1995. By Feng-Tse Lin and Cheng-Yan Kao [27] proposed a cipher text only attack on a Vernam cipher. The work by Clark, Dawson and Bergen [28] was an extension of [27]. It contains a detailed analysis of the fitness function used in [25], as well as a modified version of the same fitness function. The paper by Clark, Dawson and Nieuwland [29] is the first to use a parallel GAs for cryptanalysis. The paper published in 1997 by Clark and Dawson [30] is, overall, a slightly more detailed, longer version of [30]. This paper [31], published in 1997, is by Kolodziejczyk. It is an extension of [25], it focuses on the Merkle-Hellman knapsack system, and the effect of initial parameters on the approach reported in [25]. The paper in 1998. By Clark and Dawson, [32] compares three optimization algorithms applied to the cryptanalysis of a simple substitution cipher.

Yaseen and Sahasrabuddhe in 1999 published an important work which proposed a Gas based on the Chor-Rivest public key cryptosystem [33]. In 2002 Grundlingh and Van Vuuren, [34] combines operations research with cryptology and attacks two classical ciphers with a genetic algorithm approach.

It may be stated that the above review was carried out in accordance with the guidelines proposed by Kichenham. The review has deliberately ignored some of the papers published in grey literature. Moreover, the intend of the review was to find the pinioning works in the field, to find gaps in the existing research and to propose a new technique. The goals have largely been accomplished by the above review.

4. Analysis

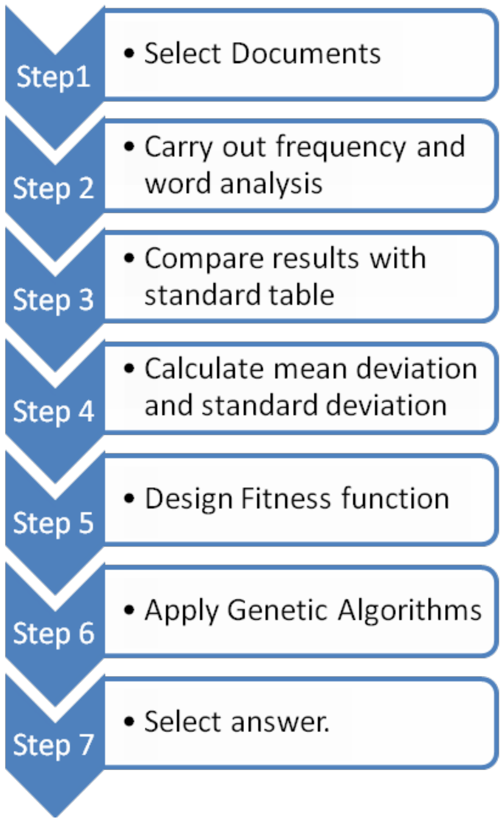

Figure 1. Analysis and procedure of Genetic Algorithm Based Cryptanalysis

The analysis of the proposed algorithm has been as follows: First of all 16 documents having length greater than equal to 2000 words are selected. These documents were then analyzed for frequency of alphabets and words in a order to generate the first table which contained the information regarding the actual frequency analysis. This table was then compared with another table which contained the standard frequency analysis of alphabets and words in a Standard English documents. This difference was then estimated by calculating the mean deviation and standard deviation of the deviations. This helped in crafting of fitness function which would assign fitness to the chromosomes of a genetic population. If we take only one key for converting plaintext to cipher text, the above analysis would convert the given problem into a optimised search problem. Hence the applicability of GAs can be justified in Cryptanalysis. The steps of the analysis and procedure can be summarized in Figure 1.

5. Proposed Work

The technique employed for the purpose of Cryptanalysis is as follows. First of all, the encrypted text will undergo a frequency analysis in which the frequency of individual token would be valid. These frequencies will be saved in a table. Henceforth, refer to as encrypted frequency table. These frequencies the given an indication of what these tokens could be do. The reason for this is that the encryption has been done of Standard English random text of around 4000 words. The frequency analysis of Standard English language text would be then used to craft the fitness function which is as follows (Refer to the appendix).

The "fitness function" is responsible for performing the evaluation and returning a positive integer number, or "fitness value", that reflects how optimal the solution is: the higher the number, the better the solution.

This is for breaking of token into simpler units. These units would again undergo a frequency analysis which would then map with a Standard English language character (Refer to the appendix).

This Fitness function would be followed by the application standard GAs. It is based on the biological evolution. In this process population is repeatedly modified and in each step the parents produce new population from the previous population which moves towards the optimized solution.

There are three main rules:

a. Selection-Selects the parents
b. Crossover-The crossover rule combines to parents to form children
c. Mutation-Changing an individual parents.

The procedure has been implemented in C#,. NET framework. The work has been then analyzed for verified using 9 text documents of about 4000 words and the results are encouraging

6. Conclusion

This paper presents a noble approach of cryptanalysis using GAs. The work has been implemented and initial results are encouraging. The analysis carried so far gave a appropriate standard deviation and minimum mean deviation. It is intended to extend the work by performing the analysis to a large datasets of around thousands of words. As a matter of fact, ways are being explored to make fitness function adaptive. In a order to do so, various machine learning approaches are applied. The work would pave the way of heuristic search algorithm in cracking keys.

References

[1] Delman, "*Genetic Algorithm in Cryptography*". Rochester Institute of Technology, Rochester, New York, 2004.

[2] Goldberg, "*Genetic Algorithm in search, Optimisation and Machine learning*", Addison Wesley Longman, London, 2006.

[3] Akpinor and Bayhan, "A Hybrid Genetic Algorithm for Mixed Model Assembly Line Balancing Problem with Parallel Workstation and Zoning Constraints", *Engineering Applications of Artificial Intelligence*, Vol, 24, pp. 449-457,2011.

[4] Al-Duwaish, "A Genetic Approach to the identification of Linear Dynamical Systems with Static Non-Linearities", *International Journal of system Science*, Vol, 31, pp, 307-313, 2000.

[5] Benjamin et al, "Genetic Algorithm using for a Batch Fermentation Process Identification", *Journal of Applied Science*, Vol. 8, pp. 2272-2278, 2008.

[6] Da Silva et al, "Genetic Algorithm with local search optimization for multiple sequences Alignment", *Applied Intelligence*, Vol. 32, pp. 164-172, 2010.

[7] Paplinski. "Genetic Algorithm with Simplex Crossover for Identification of Time Delays", Intellegent Information System, pp. 337-346, 2010.

[8] Roeva and Fidanova, "*Chapter 13 A Comparison of Genetic Algorithm and Ant Colony optimization for modelling of E. Coli Cultivation Process*", Real-world application op Genetic Algorithm, In Tech, pp-261-282, 2012.

[9] Roeva and Slavov, "*Fed-batch cultivation control based on Genetic Algorithm PID controlled Tuning*". Lecture notes on computer science, Springer-Verlag Berlin Heidelberg, Vol. 6046, pp. 289-296, 2011.

[10] Eiben Et al, "Parameter Control in Evolutionary Algorithm", *IEEE Transactions on Evolutionary Computation*, vol. 3, 1999.

[11] Nowotnaik and Kucharski, " GPU-based tuning of Quantum-Inspired Genetic Algorithm for a combinatorial optimization problem", *Bulletin of the polish Academy of Science, Technical Sciences*, Vol. 60, pp. 373-330, 2012.

[12] Saremi et al, " Tuning the Parameters of a Genetic Algorithm to solve vehicle routing Problem with Backhauls using Design of Experiments", *International Journal of Operations Research*, Vol. 4, pp. 206-219, 2007.

[13] Fidanova, "Simulated Annealing: A Monte Carlo method for GPS surveying", *Computational Science-2006, Lecture notes in computer science No. 3991*, pp. 1009-1012, 2006.

[14] Mitchell and Melanie, "*An introduction to a Genetic Algorithm*", MIT press paperback edition, 1996.

[15] Holland and John, "*Adaptation in natural and artificial systems*", A Brad Ford Book, 1992.

[16] Alander, "On optimal population size of genetic algorithm", *In Proceedings of the IEEE computer systems and software engineering*, pp. 65-69, 1992.

[17] Diaz-Gomaz and Hougen, "Initial population for genetic algorithms: A metrics approach", *In proceedings of 2007 International conference of Genetic and Evolutionary Methods, CSREA Press*, pp. 43-49, 2007.

[18] Piszcz and Soule, "Genetic Programming: optimal population sizes for varying complexity problems", *In Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 953-954, 2006.

[19] Koumousis and Katsaras, "Asaw tooth Genetic Algorithm combining the effects of variable population size and re-initialization to enhance performance", *IEEE Transaction on evolutionary computation*, vol. 10, pp. 19-28, 2006.

[20] Goldberg et al, "*Bayesuan Optimization Algorithm, population sizing and time to convergence*", Illinois genetic Algorithms laboratory, University of Illinois, USA, 2000.

[21] Lobo and Goldberg, "The parameter less genetic algorithm in practice", *Information Science, Informatics and computer science*, Vol, 167, pp. 217-232, 2004.

[22] Lobo and Lima, "A review of adaptive population sizing schemes in Genetic Algorithms", *In proceedings of the Genetic and Evolutionary Computation conference*, pp. 228-234, 2005.

[23] Spillmanet al, "Use of Genetic Algorithm in the cryptanalysis of simple substitution ciphers", *Crypto logia*, Vol. 17, pp. 31-44, 1993.

[24] Matthews and R.A.J, "The use of Genetic algorithm in cryptanalysis", *Cryptologia*, Vol. 17, pp. 187-201, 1993.

[25] Spillman, "Cryptanalysis of Knapsack Cipher using Genetic Algorithms", *Cryptologia*, Vol. 17, pp. 367-377, 1993.

[26] Clark, "Modern optimization algorithm for cryptanalysis", *In proceedings of the 1994 second Australian and New Zealand Conference on Intelligent Information system*, pp. 258-262, 1994.

[27] Lin et al, "A genetic algorithm for cipher text- only attack in cryptanalysis", *In IEEE International Conference on systems, Man and Cybernetics*, Vol. 1, pp. 650-654, 1995.

[28] Clark et al, "Combinatorial Optimization and the Knapsack cipher", *Cryptologia*, Vol 20, pp. 85-93, 1996.

[29] Clark et al, "Cryptanalysis of polyalphabetic substitution Ciphers using a parallel Genetic Algorithm*", In proceedings of IEEE International Symposium on Information and its applications*, 1996.

[30] Clark and Dawson, "A Parallel Genetic Algorithm for Cryptanalysis of Polyalphabetic Substitution Cipher", *Cryptologia*, Vol. 21, pp. 129-138, 1997.

[31] Kolodziejczyk et al, "The application of genetic algorithm in cryptanalysis of Knapsack Cipher", *In Proceeding of Fourth International Conference PRIP'97 pattern recognition and information processing*, pp. 394-401, 1997.

[32] Clark and Dawson, "Optimization Heuristics for the automated cryptanalysis of Classical Ciphers", *Journal of Combinatorial Mathematics and combinatorial computing*, Vol. 28, pp. 63-86, 1998.

[33] Yaseenaqnd Sahasrabuddhe, "A Genetic Algorithm for the cryptanalysis of chorrivest Knapsack public key cryptosystem (PKC*)", In proceedings of third international conference on Computational Intelligence and Multimedia Applications*, pp. 81-85, 1999.

[34] Grundlingh and Van Vuuren, "Using genetic algorithm to break a simple cryptographic Cipher", *Retrieved from http://dip.sun.ac.za/nvuuren/abstracts/abstr-genetic,htm, 2003.*

Appendix: Frequency of alphabets and works in a Standard English Language Text.

Table 1. Most frequently used words in English Language

The	Of	And	A
to	in	is	you
that	it	he	was
for	on	are	as
with	his	they	I
at	be	this	have
from	or	one	had
by	word	but	not
what	all	were	we
when	your	can	said
there	use	an	each
which	she	do	how
their	if	will	up
other	about	out	many
then	them	these	so
some	her	would	make
like	him	into	time
has	look	two	more
write	go	see	number
no	way	could	people
my	than	first	water
been	call	who	oil
its	now	find	long
down	day	did	get
come	made	may	part

The analysis of words in the main entries of the *Concise Oxford Dictionary* (11th edition revised, 2004) and came up with the following Table 2.

Table 2. The percentage of alphabet usage

E	11.1607%	56.88	M	3.0129%	15.36
A	8.4966%	43.31	H	3.0034%	15.31
R	7.5809%	38.64	G	2.4705%	12.59
I	7.5448%	38.45	B	2.0720%	10.56
O	7.1635%	36.51	F	1.8121%	9.24
T	6.9509%	35.43	Y	1.7779%	9.06
N	6.6544%	33.92	W	1.2899%	6.57
S	5.7351%	29.23	K	1.1016%	5.61
L	5.4893%	27.98	V	1.0074%	5.13
C	4.5388%	23.13	X	0.2902%	1.48
U	3.6308%	18.51	Z	0.2722%	1.39
D	3.3844%	17.25	J	0.1965%	1.00
P	3.1671%	16.14	Q	0.1962%	(1)