A Generic Genetic Algorithm to Automate an Attack on Classical Ciphers

Anukriti Dureha
Department of CSE
Amity School of Engg.& Technology
Amity University,Noida

Arashdeep Kaur
Department of CSE
Amity School of Engg.& Technology
Amity University, Noida

ABSTRACT

The work presented in this paper describes a generic genetic algorithm called DUREHA's (Dominance, Universal stochastic sampling and Rank-based Emulation of a Heuristic Algorithm) Algorithm for cryptanalysis of classical ciphers. The underlying objective of this paper is to automate the process of cryptanalysis in order to render salvage of time, and resources available, preserve population diversity, minimize the convergence rate and control mutation rates. While numerous algorithms have been proposed to automate this process for variegated ciphers, these approaches are yet isolated from each other. The existence of a generic algorithm to cryptanalyze any type of cipher is yet not true. The algorithm proposed in this paper aspires to address such issues. The implementation and experimentation of the proposed algorithm is accomplished using three types of classical ciphers namely mono-substitution, poly-substitution and columnar transposition. The theoretical validation and experimental results indicate that the proposed algorithm is able to decrypt the ciphers by reclaiming80.71% ,87.31%and 77.66% of letters in correct position in Mono-substitution, Columnar Transposition and Vignere cipher respectively. It is also able to distinguish between the three types of ciphers correctly and is able to correctly control the mutation andconvergence rates and preserve population diversity.

Keywords: Generic Genetic Algorithms, Multi-objective Genetic Algorithms, Diploid Genetic Algorithm Dominance, Cryptanalysis.

1. INTRODUCTION

In this age of digital information,a demand for effective internet security and a concern regarding protection of sensitive data from loss or theft is on a potent rise, since potential damages, such as alteration and elimination of crucial and confidential data can be caused. Hence, an urge to equip businesses, military and the society with effective mechanisms to enhance security of such data arises[7,14].

Cryptology[5,7] is at the heart of providing such mechanisms and furnishing such needs. It is the science of secure communication that embraces two complimentary techniques-cryptography and cryptanalysis.WhileCryptography [5,7]deals withfabricating variegated algorithms to encode and decode data in order to protect confidential information, Cryptanalysis[5,7,11]is the process of retrieving plaintext (the intended message) and/or key[5,11](functional code that transforms plaintext into its corresponding cipher text) from a ciphertext (secret message), without any authorization of the communicating parties and any prior knowledge on the key.

Cryptanalysis deals with exploiting weakness in the design of cryptographic algorithms and hence this technique is used by security experts to enhance the efficiency of the existing cryptographic algorithms and augment the robustness of their security systems. It is also used by the government to gain access to any illegal data that proves to be a big threat to national security.

Since, historical times, various cryptanalysis techniques, such as brute-force attack[11,17], frequency analysis[15] and dictionary-based attacks[4] have flourished; however, such manual tasks are not well equipped to cater to the demands of solving a cipher in minimal time in order to render it an unscathed importance. These techniques simulate a search for a solution (correct plain-text/key) through a finite, yet a vast solution space. Hence, a need to foster effective mechanisms to facilitate cryptanalysis of a cipher by procuring salvage of time and resources is intuited. These issues can be addressed by making use of Genetic Algorithms (GA) [1,11]. GA is a search optimization technique which imparts an approximated optimal solution to a specific problem. The reason behind adopting such a technique, to automate the process, is the randomness with which GA searches a solution space. This unique property induces flexibility in initiating and simulating a search through a key space which is finite, yet vast in nature.

While numerous researches have been published, giving a studyand an elucidation of the use of haploid genetic algorithms[1](simple GA) to address the key issues mentioned above, a very little emphasis have been laid upon the use of a diploid genetic algorithm. A diploid genetic algorithm [1] in artificial genetics proposes a model similar to that in natural genetics that encodes a potential solution on a double stranded chromosomal type data structure. This algorithm thrives on preserving population diversity by eradicating redundant solutions from the population. It also ensuresa controlled crossover andmutation rates, and prevents prematureconvergence of the algorithm.The existing paradigms are also very specific in nature. For example, a particular algorithm that may have been developed to cryptanalyze vignereciphers may lack the capability of breaking transposition ciphers. This indicates that, an existence of a generic algorithm imbibing the capability of cryptanalyzing any type of cipher is not true.Synthesis of a generic algorithm can be accomplished

by employing Multi-objective GAs [13]. Multi-objective or multi-criteria optimization eliminates the drawbacks of single-criterion approaches. They possess the capability of processing several criteria concurrently and conjoin them into a single number.
The research reported in this paper proposes an algorithm called DUREHA's algorithm that employs the use of multi-objective formulations and a diploid GA operator called dominance [1]. This algorithm augments the efficiency of the existing algorithms and aspires to eliminate all the drawbacks of the existing algorithms as discussed above.
The paper is organized as follows: Section II gives the literature review. Section III gives the methodology and describes the algorithm proposed. Section IV describes the experiments performed and the results obtained and finally Section V concludes the paper and gives the future scope.

2. LITERATURE REVIEW

Researches that have been reported in this area primarily accentuates on employing various meta-heuristic techniques and approaches for successful decipherment of ciphers. Various experiments have been performed to witness and analyze the type of algorithm parameters and their corresponding values to be applied to achieve greater accuracy and yield better performance. Spillman[2], for the first time, had prescribed a genetic algorithm approach to cryptanalyze substitution ciphers. This paper explores the possibility of retrieving the key by conducting a random-type search on the key space. In the same year, Spillman[3] had also accomplished success in applying a GA based approach to cryptanalyze Knapsack ciphers. Ralph Morelli et al.[4], had elucidated the expediency of a word-based genetic algorithm for solving short cryptograms. Garg[8],probed the use of genetic algorithm in breaking S-DES. In the same year, Nalini[10], gave a comparative study ,on attacking S-DES, between using GA based techniques and other optimization heuristic based techniques. Results described in this paper indicates that GA based techniques minimizes the time complexity. S.S Omran et al.[14], exhibited a study on cryptanalyzing poly substitution(vignere) cipher using GA based techniques and tested various parameters such as mutation rate and key size. The results obtained are evident of GA based approaches being more efficient.

Limitations in the existing literature: While existing work exemplifies various approaches to use genetic algorithm to attack various different type of ciphers, the approaches are yet isolated in nature. None confers a method to simulate a search that would help rendering a generic attack on all the type of ciphers. Moreover, methods prescribed so far are obscure and inaccurate in terms of specifying number of generations, i.e. some claim to obtain an optimum or an approximate of a solution in the 50th generation while others have reported a run of 109 iterations to obtain a solution. In addition to these issues, one must always keep in mind that GAs do not claim to achieve an optimal solution, rather they just aim at providing an approximation to an optimal solution. Hence this paper proposes an algorithm and a method to address and try and resolve such issues.

3. PROPOSED METHODOLOGY: Dureha's(Dominance,Universal stochastic sampling and Rank-based Emulation of a Heuristic Algorithm) Algorithm.

3.1.1. Key in the cipher text, and the algorithm parameters-Number of generation (maxgen), Population size (psize), Cross-over probability (p_c) and Mutation probability (p_m).

3.1.2. Key in the values *N,M,L*where $N+M+L=PSize_{gen}$ and *N,M,L* represents the size of keys required to be generated for mono-substitution, vignere and transposition ciphers respectively.

3.1.3. Specify the partial population size,$PSize_{gen+1}$,for every generation, that is to be obtained at the end of each generation.

3.1.4. Apply dominance operator in accordance with the following algorithm to obtain N, M, and L number of keys for each type of cipher.

3.1.4.1. Select a homologous pair of key at random.

3.1.4.2. At each locus point, the letter(allele) in the homologous pair that possesses a higher value of unigram frequency is considered to be dominant while the other letter is considered to be recessive.

3.1.4.3. At each locus point, the gene being dominant is spelled. The recessive genes are only expressed when they are accompanied by another recessive gene.

3.1.5. Decrypt the cryptogram, using all the keys generated, following the decryption algorithms described below:

3.1.5.1. ***Decryption of Mono-Substitution cipher:*** Each letter in the cipher text is matched with the cipher alphabet (key) and its corresponding plain alphabet is penned down to retrieve the plaintext from the given cipher. For Example, **Cipher text**: "kndkazjdn dap", **Decrypted Message:** "hey how are you".

Table 1.Example of Plain and Cipher alphabet.

Plain 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
Cipher Alphabet	j g w r n l b k i t f y q v a e x o c h p u z m d s

3.1.5.2. ***Decryption of Vigenere Cipher*** (Poly-substitution Cipher) :

Fig 1.VignereTabula[14]

Decryption is performed by going to the row in thevigneretablula corresponding to the letters in the key. Find the position of the cipher text letter in this row, and then use the column's label as the plaintext. An example is depicted in **Table 2.**

Table 2.Decryption Process of Vignere Cipher.

Cipher-text	B	R	F	C	S	N	S	Z	X	E	I	H
Key	U	N	I	V	E	R	S	I	T	Y	U	n
Plain-text	H	E	Y	H	O	W	A	R	E	Y	O	u

3.1.5.3. ***Decryption of Columnar Transposition:*** Break the cipher text into groups and fixed size blocks. The size of the block should be equal to the size of the cipher text divided by the inverse key length and the group size should be equal to the key length. Transpose the matrix formed after breaking the cipher text into groups and fixed size blocks. For example if the inverse key p' is {7,1,2,9,3,5,0,4} of key length 8 and the cipher text is: "rsdhaecxarreitmmistcfel@pibogseaetngaeroomhslfia" of size 48, the size of the block = (48/8) 6,while the group size=8,as depicted below in Table 3a and 3b.An extra letter(say"@") can be padded at the end of the cipher-text if the size of the message is not a multiple of the key size, which is equal to 8in this case.

Table 3a.Example depicting Encryption Process of Transposition Cipher

7	R	S	D	H	A	E
1	C	X	A	R	R	E
2	I	T	M	M	I	S
9	T	C	F	E	L	@
3	P	I	B	O	G	S
5	E	A	E	T	N	G
0	A	E	R	O	O	M
4	H	S	L	F	I	a

Table 3b. Transpose of matrix formed in the table above

Z	1	2	9	3	5	0	4
R	C	I	T	P	E	A	h
S	X	T	C	I	A	E	S
D	A	M	F	B	E	R	L
H	R	M	E	O	T	O	F
A	R	I	L	G	N	O	I
E	E	S	@	S	G	M	a

Rearranging the columns in the order,{0,1,2,3,4,5,7,9}, and reading off the text row wise, we get a plain text as"aciphertextisascrambledformoftheorignalmessage,

3.1.6. Compute fitness of each key using Eq. 1,depicted below:

$$Fit(K) = 1 - ((\alpha * \sum_{i\epsilon A} |L_{(i)}^{u.s} - M_{(i)}^{u.s}| + \beta * \sum_{i,j\epsilon A} |L_{(i,j)}^{b.s} - M_{(i,j)}^{b.s}| + \gamma * \sum_{i,j,k\epsilon A} |L_{(i,j,k)}^{t.s} - M_{(i,j,k)}^{t.s}|)/x) \quad (1)$$

Here, x is some constant,*A* denotes the language alphabet i.e., for English,[A. . . Z], *L* and *M* denote the known language statistics and decrypted message statistics, respectively, and the indices *u.s*, *b.s* and *t.s* denote the unigram, bigram and trigram statistics, respectively. The values of α, β and γ are used to assign different weights to each of the three n-gram types, such that $\alpha + \beta + \gamma = 1$, and are determined experimentally.

3.1.7. Using multi-objective optimization,compute an aggregate fitness over the three types of fitness values produced in step 3.1.6.

3.1.7.1. Compute the rank of a key K in generation Genby using the formula in Eq. 2**.**

$Rank(K, Gen)$=1+ count of fit_K^{Gen}.(2)

fit_K^{Gen}represents the fitness of key *K* in generation Gen.

3.1.7.2. Assign a fitness value to each key *K* based on its rank in generation Gen in accordance with the formula in Eq.3

$$f(K, Gen) = 3 * PSize_{Gen} - \sum_{i=1}^{Rank(K,Gen)-1} Count\ of\ Rank(i) - 0.5 * count\ of\ rank(K, Gen) - 1 \quad (3)$$

3.1.7.3. Compute the Euclidean distance between every solution pair *K* and K' by using the formula in Eq. 4.

$$Eu_{Dis(K,K')} = \sum_{i=1}^{NCA} \left(fit_i(K) - fit_i(K') \Big/ fit_i^{max} - fit_i^{min} \right)^2 \quad (4)$$

3.1.7.4. Compute the niche size(σ_{share}) using the formula in Eq. 5:

$$\sigma_{share} = \sqrt{\frac{\sum_{i=1}^{3*PSize_{Gen}}\left(fit_i - \frac{\sum_{i=1}^{3*PSize_{Gen}} fit_i}{3*PSize_{Gen}}\right)}{3*PSize_{Gen}}} \quad (5)$$

3.1.7.5. Compute the Niche count*(NC(K,Gen))* of each $K \in P_{Gen}$, as described in Eq.6.

$$NC(K, Gen) = \sum_{K' \in P_{Gen}, Rank(K',Gen)=Rank(K,Gen)} max\left\{\frac{\sigma_{share} - Eu_{Dis(K,K')}}{\sigma_{share}}, 0\right\} \quad (6)$$

3.1.7.6. Compute the shared fitness $f'(K, Gen)$ for each K$\in P_{Gen}$ in accordance with Eq. 7

$$f'(K, Gen) = {f(K, Gen)}\Big/{NC(K, Gen)} \quad (7)$$

3.1.7.7. Normalize the fitness values using the shared fitness values in accordance with Eq.8

$$f''(K, Gen) = \frac{f'(K,Gen) * count\ of\ rank(K,Gen)}{\sum_{K' \in P_{Gen}, Rank(K',Gen)=Rank(K,Gen)} f'(K,Gen)} * f(K, Gen) \quad (8)$$

3.1.8. Using Universal Stochastic Sampling, select $PSize_{gen}$ number of keys.

3.1.9. Perform crossover with crossover probability p_c and mutation with mutation probability p_m, taking any two keys at random. Place the newly generated off -springs in the new population. Continue until the size of new population is equal to$PSize_{gen}$.

3.1.10. Stop if there's no further improvement in the fitness values.

4. EXPERIMENTS ANDRESULTS

The experimentation was done using C++ and STL with 2 GB Ram, Intel i5 processor on Microsoft Windows 7 Ultimate.The proposed algorithm was tested by initializing *N, M and L* with equal values.
Three experiments were conducted on the three types of ciphers to evaluate the algorithm. These experiments are as follows:

Experiment 1:Values of α, β, and γ were varied against the percentage of correct words retrieved for different values of cipher text length and key length, keeping all other parameters constant. The experimental results are described in Table 4. The results obtained indicates that, for Mono-Substitution ciphers the values of α were greater than the values of β and γ. Conversely, if the values of α were greater than the values of β and γ, the cipher-text was found to be encrypted with Mono-Substitution encryption algorithm.For Vignere cipher, the values of βwere greater than the values of α and γ. Conversely, if the values of β were greater than the values of α and γ, the cipher-text was found to be encrypted with vignere encryption algorithm.For columnar transposition ciphers, the values of γ were greater than the values of α and β. Conversely, if the values of γ were greater than the values of α and β, the cipher-text wasfound to be encrypted with columnar transposition encryption algorithm.

Table 4. Results of Experiment1

Encryption-Algorithm	α	β	γ	**% of correct letters**	**Correct Algorithm retrieved?**
Mono-Substitution	0.2	0.5	0.3	21.31	No
Mono-Substitution	0.3	0.6	0.1	35.03	No
Mono-Substitution	0.5	0.4	0.1	75.63	Yes
Mono-Substitution	0.6	0.35	0.05	80.71	Yes
Columnar-Transposition	0.01	0.19	0.8	87.31	Yes
Columnar-Transposition	0.15	0.23	0.62	83.76	Yes
Columnar-Transposition	0.5	0.4	0.1	31.98	No
Columnar-Transposition	0.6	0.35	0.05	29.95	No
Vignere	0.35	0.15	0.5	15.23	No
Vignere	0.3	0.26	0.39	25.89%	No
Vignere	0.2	0.7	0.1	67.01%	Yes
Vignere	0.21	0.76	0.03	77.66%	yes

Experiment 2: Cipher-text length, key length, number of generations and the population size were varied against the percentage of correct words retrieved, keeping all other parameters constant.Figure.2 shows that for vignere cipher, when cipher-text length was equal to 86 letters, 50% of correct letters were obtained for key-length = 6letters while only 34.88% of correct letters were obtained when key length =19 letters. For cipher-text length=197, 77.66% of correct letters were retrieved when key-length =19 and 55.84% with key-length=6. For columnar transposition ciphers,when cipher-text length = 86,78% of correct letters were retrieved with key-length=6, while only 46.51% correct letters were retrieved with key-length=19. Whereas, for cipher-text length=197, 74.62% correct letters were retrieved with key-length=6, while 87.31% correct letters were retrieved with key-length=19. Similarly, for mono-substitution ciphers, when the cipher-text length=86, 63.95% of correct letters were retrieved with key-length=6 while only 37.20% of correct letters were retrieved with key-length=19. Whereas, for cipher-text length=197, 64.97% of correct letters were retrieved with key-length=6, while 80.71% of correct letters were retrieved with key-length=19.These results indicate that if the cipher-text length is small, better results are produced with smaller key-size;

however, if the cipher-text length is relatively large, better results are produced with keys of larger key-length. If the key –length is relatively less as compared to the cipher text-length, better results can be obtained by using a bigger population size.

.

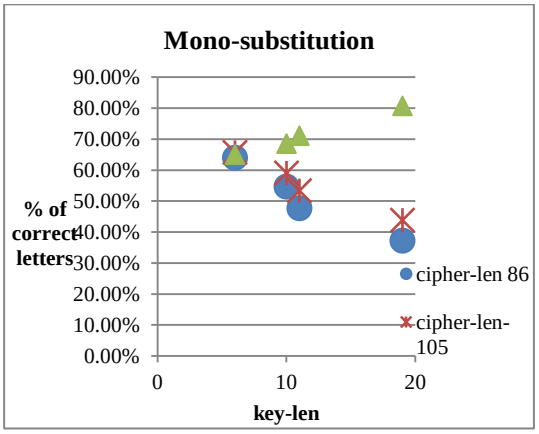

Fig2 a: Key-length vs. % of correct letters for Mono-Substitution cipher.

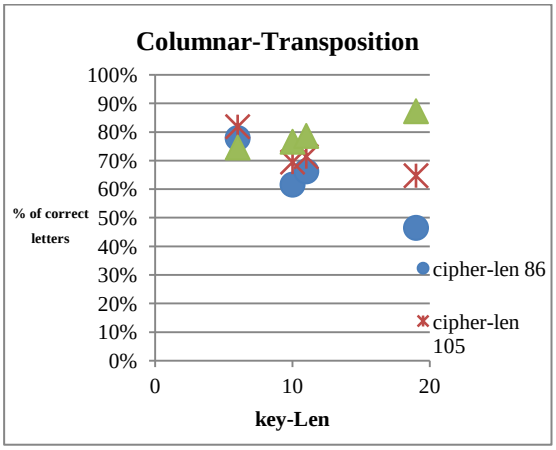

Fig2 b:Keylenvs % of Correct letters for Mono-Substitution Cipher.

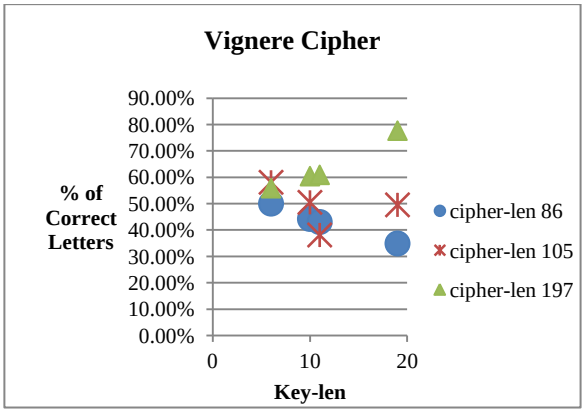

Fig2 c: Keylenvs % of correct letters for Columnar-Transposition cipher.

Experiment 3: Mutation probability was varied against the no. of generations, keeping all other parameters constant as depicted in Figure3 . The graph in Figure 3 shows us that, the least number of generation (=19 generations) for simulation of the algorithm, was obtained at p_m =0.0212 for Mono-substitution ciphers, at p_m =0.02 for Columnar transposition while p_m =0.052 for vignere cipher. The values of mutation probability are less as compared to the ones in the existing literature suggesting a controlled mutation rate.

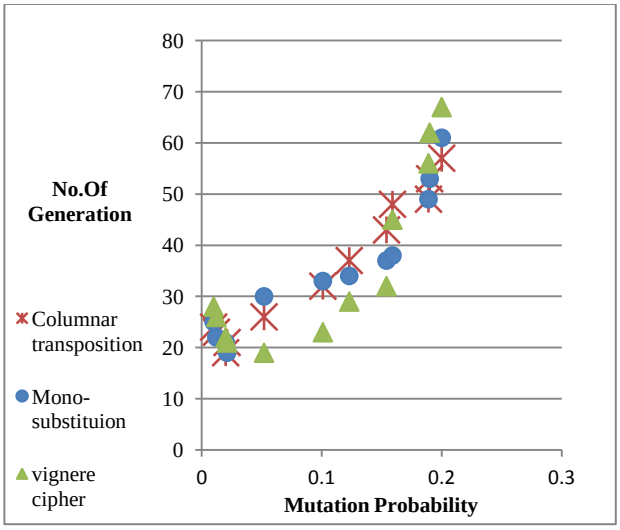

Fig 3: Mutation probability vsNo.of generations

The above experimental results also elucidate the fact that, by using optimum mutation rates, the number of generations does not exceed a total of 50 generations. This is a significant improvement over the existing algorithms as it helps in reducing the overall computation time. Moreover, since the algorithm employees the use of a reduction size in each generation, accuracy of results was also enhanced. The results also illustrates that, out of the three types of ciphers, Transposition ciphers produce the best possible results.

5. CONCLUSION AND FUTURE SCOPE

The results indicate that the proposed algorithm was successful in controlling mutation rates and hence in preserving Population Diversity. It was also successful in retrieving most of the bits(80.71% ,87.31%,77.66%for Mono-substitution,Columnar Transposition and Vignere cipher respectively)in correct position and the type of the cipher algorithm it was encrypted with originally. By using, a significant reduction size in each generation, accuracy of results was also enhanced.The results also indicate that number of bits retrieved is directly proportional to the key-length relative tothe cipher-textlength. It is also directly proportional to the population size as it helps in preserving population diversity.The overall computation time was also reduced by reducing the number of generations to less than 50 generations which helped in instilling a higher robustness into the algorithm. The proposed algorithm is a generic algorithm and demands no prior knowledge on any specific

property of a cipher for its simulation and hencecan be used to cryptanalyze any type of a given cipher.

The algorithm proposed in this paper is a promising and an effective method to facilitate businesses and the society with effective security mechanisms that will help them in keeping their data secured. It will also arm the military with efficient mechanisms that will help them in gainingaccess to any illegal data causing a threat to the national security.

Our Plans for future work aims at improvising the algorithm further to achieve a higher accuracy for the system, testing and implementation of the proposed attack for other classical and modern ciphers, for example, AES and DES .

Since, English is the only language taken into consideration while developing this tool, plans to incorporate mechanisms for other languages is also anticipated.

6. REFERENCES

[1]. David E Goldberg,"Genetic algorithms in search, optimization and machine learning", Addision- Wesley Pub.Co.1989.

[2] SpillmanR,Janssen M, Nelson B and Kepner N, "Use of Genetic Algorithm in Cryptanalysis of Simple Substituion Cipher" Cryptologia, Vol.17, No.4, pp. 367-377, 1993.

[3]. SpillmanR,"Cryptanalysis of Knapsack Ciphers using Genetic Algorithms", Cryptologia, Vol.17, No.4, pp. 367-377, 1993.

[4]. Ralph Morelli,RalphWalde,"A word-based genetic algorithm for cryptanalysis of short cryptograms",Flairs,2003.

[5]. Mao, W., "Modern Cryptography: Theory & Practice." Upper Saddle River, NJ: Prentice Hall PTR, 2004.

[6]. Eng. Ayman M, B. Albassal', Prof. Dr. Abdel-Moneim A. Wahdan2, "Genetic algorithm cryptanalysis of the basic substitution permutation network",ieee,2004.

[7]. PoonamGarg, AdityaShastri, and D.C. Agarwal," An Enhanced Cryptanalytic Attack on KnapsackCipher using Genetic Algorithm", World Academy of Science, Engineering and Technology 12 2005.

[8]. GargPoonam, "Genetic algorithm Attack on Simplified Data Encryption Standard algorithm", International journal Research in Computing Science, ISSN1870-4069, 2006.

[9]. Abdullah Konak, David W. Coit, Alice E. Smith," Multi-objective optimization using genetic algorithms": A tutorial,Elsevier, Reliability Engineering and System Safety, 9 January 2006.

[10]. Nalini, "Cryptanalysis of Simplified data encryption standard via Optimization heuristics", International Journal of Computer Sciences and network security, vol 6, No 1B, Jan 2006.

[11]. R.Toemeh,S.Arumugam,"Breaking Transposition Cipher with Genetic Algorithm", Electronics and Electrical Engineering,ISSN 1392 – 1215,2007.

[12]. GargPoonam, "Memetic Algorithm Attack on Simplified Data Encryption Standard Algorithm", proceeding of International Conference on Data Management, February 2008, pg 1097-1108 .

[13]. Tania Pencheva, KrassimirAtanassov, Anthony Shannon, "Modelling of a Stochastic Universal Sampling SelectionOperator in Genetic AlgorithmsUsing Generalized Nets", Tenth Int. Workshop on Generalized Nets Sofia, 5 December 2009.

[14]. S. S. Omran ,A. S. Al-Khalid D. M., Al-Saady," A Cryptanalytic Attack on Vigenère Cipher Using Genetic Algorithm",IEEE conference on Open systems,2011.

[15] Rod Hilton, "Automated Cryptanalysis of MonoalphabeticSubstitution Ciphers Using Stochastic OptimizationAlgorithms"-thesis.

[16]. JitinLuthra,Saibal K. Pal,"A Hybrid Firefly Algorithm using Genetic Operators for the Cryptanalysis of aMonoalphabetic Substitution Cipher",IEEE,2011.

[17]. Vimalathithan. R., M. L. Valarmathi, European Journal of Scientific Research," Cryptanalysis of DES using Computational Intelligence", ISSN 1450-216X Vol.55 No.2, 2011.

AUTHOR'S PROFILE

Anukriti Dureha is associated with the Department of Computer Science and Engineering, Amity School of Engineering and Technology, NOIDA, India. She completed her IGCSE(International General Certificate of Secondary Education) at Legae Academy, Gaborone, Botswana. She is currently pursuing B.Tech in Computer Science and Engineering from Amity University, Uttar Pradesh, India.

Arashdeep Kaur is associated with the Department of Computer Science and Engineering, Amity School of Engineering and Technology, NOIDA. She did her B.Tech in Computer Science and Engineering with honors from Punjab Technical University, Jallandhar, India and M.Tech in Computer Science and Engineering from Guru Nanak Dev Engineering College, Ludhiana, Punjab, India. Currently, she is pursuing Ph.D. from Amity University, Uttar Pradesh, India in the field of multimedia security and digital watermarking.