Metaheuristic Techniques for Automated Cryptanalysis of Classical Transposition Cipher: A Review

Chapter · January 2022

DOI: 10.1007/978-981-16-2877-1_43

5 authors, including:

Santosh Kumar Vishwakarma
Manipal University Jaipur
43 PUBLICATIONS **315** CITATIONS

SEE PROFILE

Nirmal Gupta
Jaypee University Anoopshahr
22 PUBLICATIONS **105** CITATIONS

SEE PROFILE

Vaibhav C. Gandhi
Navrachana University - Vadodara
10 PUBLICATIONS **9** CITATIONS

SEE PROFILE

Metaheuristic Techniques for Automated Cryptanalysis of Classical Transposition Cipher: A Review

Ashish Jain, Prakash C. Sharma, Santosh K. Vishwakarma, Nirmal K. Gupta
School of Computing and Information Technology
Manipal University Jaipur, Jaipur, India
{ashish.jain, prakashchandra.sharma, santosh.kumar,
nirmalkumar.gupta}@jaipur.manipal.edu
Vaibhav C. Gandhi
Department of Computer Science and Engineering
Navrachana University, Vadodara, India
vaibhavgandhi2424@gmail.com

Abstract. Between the year 1994 and 2018, a considerable new and different metaheuristic optimization techniques have been presented in the literature for automated cryptanalysis of classical transposition cipher. This paper compares the performance of these new and different metaheuristic techniques. Three main comparison measures are considered to assess the performance of presented metaheuristics: effectiveness, efficiency, and success rate. It is noteworthy that among the presented metaheuristics the performance of genetic algorithm technique is best with respect to all the measures.

Keywords: Genetic Algorithm, Simulated Annealing, Tabu Search, Cryptanalysis.

1. Introduction

Combinatorial optimization is an approach to deal with a given problem and locate the best answer out of a very large set of possible solutions. The problems for which one need to find the best solutions are mostly comes under the umbrella of NP-hard and NP-complete combinatorial problems. The problem associated related to solving these problems is that the time and/or memory increases drastically with the increase in size of problems [1]. Branch and bound and simplex methods are examples of exact optimization techniques that can be used to speed up the search. However, often these techniques have prohibitive complexity requirements (time and/or memory) which makes the use of these techniques impractical [2]. In such cases, approximate techniques, i.e., metaheuristics are utilized to determine an adequate solution to the problem [2]. This paper presents three different metaheuristic techniques in solving the problem related to the classical transposition cipher.

The concept of Turing Machine (TM) and the class of decision problems are often used to understand the theory of NP-completeness. "yes" or "no" these are the two

possible solutions that are associated with a decision problem [1]. In polynomial time the deterministic TM can solve a set of problems, let such problems categorized in a set P. Similarly, in polynomial time the non-deterministic TM can solve a set of problems, let such problems categorized in a set NP. Let P ⊆ NP (see Figure 1). When a decision problem belongs to both in NP and NP-hard (hard problems) such problems comes in the category of NP-completeness. Formally, a decision problem X is NP-complete if X satisfy the following two conditions [1]:

(i) X is in NP, and (ii) Every problem in NP is "reducible" to X in polynomial time If a candidate solution of X can be verified in polynomial time, then we can say that X is in NP [1]. Note that whether a problem satisfies condition (i) or not, but if it satisfies condition (ii) then said to be NP-hard problem [1]. The cryptanalysis problems that are considered to solve in this paper are exists in the class of NP-complete problems.

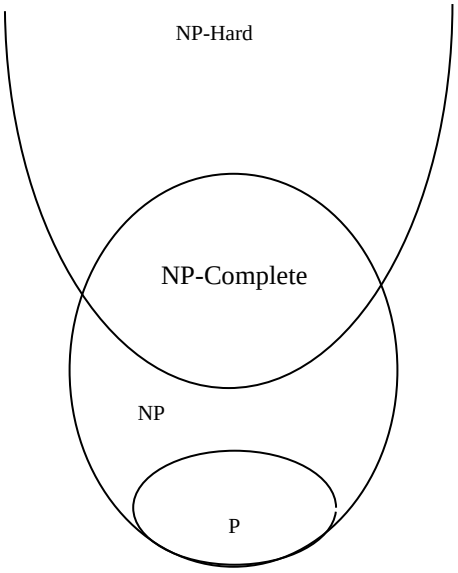

Figure 1: P, NP, NP-complete and NP-hard [1]

1.1 Automated Cryptanalysis

Cipher provides information security. Basically, ciphers are used to transform one form of text called “plaintext” into another form of text called “ciphertext” which is tough to break if the secret key is not known. Cryptanalysis is the process of finding weakness in the design of the ciphers. Cryptanalyst performs cryptanalysis. One of the most difficult tasks of the cryptanalyst is to discover (detect or search) the secret key of the cipher by knowing only some of the ciphertext characters. In terms of information security if cryptanalyst or attacker able to discover the secret key of the cipher then we

say that the cipher has been successfully attacked. Attacking cipher exists in the class of NP-complete problem [3-9]. If the exhaustive search is carried out to detect secret key in the keyspace, then the whole keyspace required to be examined in the worst case that will take significant number of years [3-10]. However, automated attacks can be formed using metaheuristic techniques that can search the secret key of classical ciphers in acceptable amount of time [5, 6, 9, 10].

1.2 The Classical Transposition Cipher

Here we describe three components related to the transposition cipher: secret key, encryption procedure and decryption procedure. A permutation sequence is used to represent secret key. This key is used to convert plaintext into ciphertext. The process of conversion is as follows: first, the fixed size segments are formed of the given plaintext, then on each segment apply the same sized permutation key (for example, see Table 1).

Table 1: Encryption of a plaintext: an example

Plaintext	Character position	Permutation key	Ciphertext
T	1	4	_
H	2	1	T
E	3	5	S
_	4	6	T
S	5	2	H
T	6	3	E
U	1	4	N
D	2	1	U
E	3	5	T
N	4	6	S
T	5	2	D
S	6	3	E
_	1	4	R
W	2	1	_
E	3	5	E
R	4	6	_
E	5	2	W
_	6	3	E

Consider the following permutation sequence: $\pi = \{4, 1, 5, 6, 2, 3\}$. Since permutation sequence sized is six, therefore the plaintext is segmented in the size of six (see Table 1). That is, upon encryption the 4th letter will come at location 1st. 1st will come at 2nd. 5th will come at 3rd. 6th will come at 4th. 2nd will come at 5th, and 3rd will come at location 6th. The inverse process will be applied for decryption, i.e., $\pi^{-1} = \{2, 5, 6, 1, 3, 4\}$. Here, the noteworthy point is that the recipient must know the secret key for decryption.

The remaining parts of the paper are arranged as follows: in Section 2 three criteria for performance measurement of the metaheuristic are described. In Section 3, a literature review has been carried out. In Section 4, the results obtained using metaheuristics are discussed. In Section 5, conclusion of the paper has been summarized.

2. Performance Measurement Criteria

The transposition cipher was discussed in the section 1.2. One can ask what the weakness in the cipher is so that it can be attacked. The answer is – the encryption process used in the transposition cipher does not altered the character frequency distribution significantly. Therefore, the metaheuristics are capable to match the known language statistics with the character frequency statistics (n-grams) of the encrypted message (a standard strategy to automatically attack the classical ciphers).

There are three criteria based on which the performance of metaheuristic techniques can be assessed with regard to automated attacks: (1) number of ciphertext characters available for the attack (effectiveness measurement criterion); (2) number of key elements detected correctly (success rate measurement criterion); (3) time required to recover the key (efficiency measurement criterion). Based on these three main criteria we will assess the performance of different metaheuristic techniques in the result section.

3. Literature Review

Automated attacks are run without tedious connection of people with the search procedure and ends when the secret key is detected [9, 10]. The application of metaheuristic techniques in automated attacks of classical transposition ciphers was first reported in 1994 (e.g., [11]-[12]), and the outcomes have demonstrated that metaheuristic strategies are exceptionally efficient and effective. With this inspiration, numerous metaheuristic techniques have been reported for mounting automated attacks on the transposition cipher, for example, genetic algorithm, simulated annealing, and tabu search. Among all these algorithms the genetic algorithm recently proposed by Jain and Chaudhary [13] has shown the best performance with respect to efficiency and success rate.

For automated cryptanalysis of the “classical transposition cipher, hereinafter,

transposition cipher" multiple metaheuristic techniques have been used in the past that have been mentioned above. Below we describe the standard form of these techniques in brief.

In 1960s, Holland and his students [14, 15] proposed a popular population-based metaheuristic, namely, genetic algorithm. This method starts by haphazardly creating a population of individuals. Three operators, namely, selection, crossover, and mutation control the population and to generate the new population from the old population. In each generation a cost function, namely, fitness function assesses the suitability of individuals. After some number of iterations, the individual with best cost provide the adequate answer to the associated issue. For point by point depiction on the genetic algorithm the reader can refer [14-17].

Simulated annealing is a metaheuristic technique that handle and updates a single solution during optimization. Kirkpatrick et al. [18] mimicked the annealing process with respect to combinatorial optimization. The simulated annealing strategy starts with a haphazard solution for the issue to be solved and a beginning temperature. At every temperature various endeavour are made to bother the current solution. For point by point depiction on the simulated annealing the reader can refer [18].

Tabu search is a direction-based metaheuristic technique which gives a way to deal with search to find an ideal arrangement of the given issue. A separate list, namely, a tabu list is preserved by the tabu strategy during the hunt of the solution [19]. The additional arrangement stays in the tabu for a characterized number of iterations. For point by point depiction on the tabu search the reader can refer [19, 20].

In the literature, genetic algorithm, simulated annealing, and tabu search have been utilized to tackle many optimization issues. These methods have also utilized for the optimization problems related to cryptology. The cryptology problems solved using these methods and their applications is shown in Table 2.

Table 2: Applications of cryptology problems solved using genetic algorithm, simulated annealing and tabu search

Authors [Reference]	Metaheuristics Used	Problem Solved	Application
Jain and Chaudhari [9] Matthews [21] Spillman et al. [22] Clark [12, 23] Dimovski and Gligoroski [24] Garg and Sherry [25] Verma et al. [26] Omran et al. [27] Mudgal et al. [28]	Genetic Algorithm	Automated Cryptanalysis of Classical Substitution Cipher	Modern Substitution Ciphers Uses Functions of Classical Substitution Cipher in a Complicated Way [8]
Garici and Drias [29]	Scatter Search		

Forsyth and Naini [30] Clark [12, 23]	Simulated Annealing		
Clark [12, 23] Garg and Sherry [25] Verma et al. [26]	Tabu Search		
Giddy and Safavi-Naini [11] Clark [12, 23] Jain and Chaudhari [13] Toemeh and Arumugam [31] Song et al. [32] Muhajjar [33] Al-Khalid et al. [34] Garg [35]	Genetic Algorithm	Automated Cryptanalysis of Classical Transposition Cipher	Modern Transposition Ciphers Uses Functions of Classical Transposition Cipher in a Complicated Way [8]
Giddy and Safavi-Naini [11] Clark [12, 23] Song et al. [32] Garg [35] Mishra and Kaur [36]	Simulated Annealing		
Clark [12, 23] Garg [35]	Tabu Search		
Clark [23] Spillman [37] Yaseen and Sahasrabuddhe [38] Garg et al. [39] Ramani and Balasubramanian [40]	Genetic Algorithm	Automated Cryptanalysis of Knapsack Cipher	Knapsack Cipher is a Reasonable Alternative, Particularly for Security of Little Implanted Gadgets, e.g., Cellular Devices [8]
Song et al. [41] Vimalathithan and Valarm-athi [42] Sathya et al. [43] Sharma et al. [44] Al Adwan et al. [45] Dworak and Boryczka [46]	Genetic Algorithm	Automated Cryptanalysis of Data Encryption Standard (DES)	DES is a Modern Block Cipher Used for Encryption of Confidential Information [8]
Nalini and Rao [47] Nalini [48]	Simulated Annealing		

Nalini and Rao [47] Soyjaudah [49]	Tabu Search		
Cowan [50]	Simulated Annealing	Automated Cryptanalysis of Short Playfair Ciphers	Modern Substitution Ciphers Uses Functions of Playfair Substitution Cipher in a Complicated Way [8]
Clark et al. [51]	Simulated Annealing	Design of Substitution-boxes	Substitution-boxes are Nonlinear Elements that are Used in Block Cipher for Encryption of Confidential Information [8]

4. Comparative Analysis

Recall from Section 2, the standard strategy for escalating attacks on the transposition cipher is the matching of the known language statistics with the observed n-gram statistics of the decrypted message. Through matching we determined the cost of the candidate key. A candidate key is a key which is evolved using metaheuristic technique during the hunt of original secret key.

Fitness Function. The input of this function is the candidate key. This function determines the "quality" of the candidate key. For example, from the population of the evolved candidate keys, a key *K* is selected. Using *K*, a known ciphertext is decrypted. Afterwards, an examination is carried out between n-gram statistics of the decoded ciphertext and the known language statistics (for instance, for English language statistics refer [9]). Thusly, the fitness of *K* is determined. Formally, Eq. (1) is utilized for statistics comparison.

$$Cost_k = \alpha(\sum_{i\in\zeta} |k_i^u - d_i^u|) + \beta(\sum_{i,j\in\zeta} |k_{i,j}^b - d_{i,j}^b|) + \gamma(\sum_{i,j,k\in\zeta} |k_{i,j,k}^t - d_{i,j,k}^t|) \quad (1)$$

For clarification on Eq. (1) the reader can refer [9]. The estimation of n in the term n-gram ought to be higher in number to play out a precise appraisal of candidate keys. In any case, in the writing it has demonstrated that typically the best operational reason for a fitness function utilized in automated cryptanalysis of transposition ciphers are the bigrams and trigrams only [13]. Instead of using all possible bi/trigrams, an alternative approach is to use a subset of most common bi/trigrams for efficient valuation of a candidate key. For example, Table 3 can be used to calculate the cost of a candidate key. The fitness function using the weight table like Table 3 can be represented using Eq. (2). For clarification on Eq. (2) the reader can refer [13].

$$Cost_k = \sum_{i\in\eta} F_i S_i\,, \quad (2)$$

Table 3: The *n*-gram ($n > 1$) weight table used in this research

bigrams	score	bigrams	score	trigrams	score
E_	+2	_A	+1	_TH	+5
_T	+1	S_	+1	HE_	+5
HE	+1	- -	-6	THE	+5
TH	+1			- - -	-10

Experiment. For performing experiments, the considered metaheuristic techniques have been implemented in Java. We followed the guidelines reported in the respective papers during implementation of each of the metaheuristics. Intel Quad-Core processor i7 (@3.40Ghz) is used to execute the presented metaheuristics. Given ciphertext, length of the ciphertext (1000 characters), and the weight Table 3 are input to every algorithm. Afterwards, the performance of every algorithm has been assessed on hundred distinct known ciphertexts. For cryptanalysis purpose, we have taken the distinct messages from various magazines and storybooks randomly.

Analysis of Results. Regarding all the performance criteria, we mention the obtained results in the Table 4. Note that the metaheuristic technique that takes a greater number of ciphertext characters are said to be less effective than the metaheuristic which takes lesser number of ciphertext characters. From the obtained results, we can observe that all the algorithms are equally effective because taking 1000 number of ciphertext characters for successful recovery of key. From the obtained results, we can clearly observe that the genetic algorithm proposed by Jain and Chaudhari [13] takes only 1000 ciphertext characters and as an outcome able to recover 21.11 number of key elements

out of 25. The time taken by the algorithm is also less (5.23 seconds) as compare to other algorithms. This study indicates that the genetic algorithm is most efficient and most successful in escalating attacks on the transposition cipher.

Table 4: Cryptanalytic results obtained through various metaheuristic techniques (transposition size = 25)

Year	Authors [Reference]	Metaheuristics Used	Maximum Number of Ciphertext Characters Used	Average Number of Key Elements Correctly Recovered out of 25	Mean Performance Time (in seconds) to recover the key
1994	Giddy and Naini [11]	Genetic Algorithm	1000	20.24	6.05
1994	Giddy and Naini [11]	Simulated Annealing	1000	20.31	6.02
1994, 1998	Clark [12, 23]	Genetic Algorithm	1000	20.63	6.16
1994, 1998	Clark [12, 23]	Simulated Annealing	1000	20.71	5.98
1994, 1998	Clark [12, 23]	Tabu Search	1000	20.83	5.84
2007	Toemeh and Arumugam [31]	Genetic Algorithm	1000	20.82	6.64
2008	Song et al. [32]	Genetic Algorithm	1000	21.09	6.92
2008	Song et al. [32]	Simulated Annealing	1000	20.73	6.01
2009	Garg [35]	Genetic Algorithm	1000	20.67	6.12
2009	Garg [35]	Simulated Annealing	1000	20.73	5.96
2009	Garg [35]	Tabu Search	1000	20.85	5.81
2010	Muhajjar [33]	Genetic Algorithm	1000	20.69	6.09
2013	Al-Khalid et al. [34]	Genetic Algorithm	1000	20.72	6.07

2015	Mishra and Kaur [36]	Simulated Annealing	1000	20.75	5.93
2018	Jain and Chaudhari [13]	Genetic Algorithm	1000	21.11	5.23

5. Conclusions

The efficient, effective, and successful utilization of various metaheuristic techniques in solving the transposition cipher is presented. Based on the results presented in Table 4, we noted the following performance with respect to key recovery and the time taken: (1) The performance of simulated annealing proposed by Giddy and Naini [11] is better as compared to his genetic algorithm. (2) The performance of tabu search proposed by Clark [12, 23] is best as compared to his genetic algorithm and simulated annealing. (3) The performance of genetic algorithm proposed by Song et al. [32] is better as compared to his simulated annealing algorithm. (4) The performance of tabu search proposed by Garg [35] is best as compared to his genetic algorithm and simulated annealing. (5) If we compare the performance of simulated annealing among all the proposed simulated annealing algorithms by different authors, then we can say that the performance of simulated annealing prosed by Mishra and Kaur [36] is best. (6) If we compare the performance of tabu search of Clark [12, 23] and Garg [35], then we can say that the performance of tabu search proposed by Garg [35] is little bit better. However, the performance of genetic algorithm recently proposed by Jain and Chaudhari [13] is extremely better than the all the previously proposed algorithms. This study indicates that the genetic algorithm technique proposed in [13] is a viable option for solving such kind of NP-complete problems.

References

1. Goldreich, O. (2010), P, NP, and NP-Completeness: The Basics of Computational Complexity. Cambridge University Press, pp. 1-183.
2. Du, K. L., & Swamy, M. N. S. (2016), Search and Optimization by Metaheuristics: Techniques and Algorithms Inspired by Nature, Birkhäuser, pp. 1-434.
3. Menezes, A. J., Van Oorschot, P. C., & Vanstone, S. A. (1996), Handbook of Applied Cryptography, CRC press, pp. 1-780.
4. Stinson D. R. (2005), Cryptography: Theory and Practice, CRC press, pp. 1-593.
5. Castro, J. C. H. and Viñuela, P. I. (2005), Evolutionary Computation in Computer Security and Cryptography, New Generation Computing, 23 (3), pp. 193-199.
6. Danziger, M., & Henriques, M. A. A. (2012), Computational intelligence applied on cryptology: a brief review. IEEE Latin America Transactions, 10(3), pp. 1798-1810.

7. Awad, W. S., & El-Alfy, E. S. M. (2015), Computational Intelligence in Cryptology. Improving Information Security Practices through Computational Intelligence, vol. 28, pp. 1-17.
8. Holden, J. (2017), The Mathematics of Secrets: Cryptography from Caesar Ciphers to Digital Encryption. Princeton University Press, pp. 1-373.
9. Jain, A., & Chaudhari, N. S. (2019). An Improved Genetic Algorithm and A New Discrete Cuckoo Algorithm for Solving the Classical Substitution Cipher. International Journal of Applied Metaheuristic Computing (IJAMC), 10(2), 109-130.
10. Bhateja, A. K., Bhateja, A., Chaudhury, S., & Saxena, P. K. (2015), Cryptanalysis of vigenere cipher using cuckoo search. Applied Soft Computing, 26, 315-324.
11. Giddy, J. P., & Safavi-Naini, R. (1994). Automated cryptanalysis of transposition ciphers. The Computer Journal, 37(5), 429-436.
12. Clark, A. (1994), Modern optimisation algorithms for cryptanalysis, In IEEE proceedings of the Intelligent Information Systems 1994, IEEE, pp. 258-262.
13. Jain, A., & Chaudhari, N. S. (2018). A novel cuckoo search technique for solving discrete optimization problems. International Journal of System Assurance Engineering and Management, 9(4), 972-986.
14. Goldberg, D. E. (2006). Genetic algorithms. Pearson Education India.
15. Michalewicz, Z. (2013). Genetic algorithms+ data structures= evolution programs. Springer Science & Business Media.
16. Gonzalez, T. F. (Ed.). (2007). Handbook of approximation algorithms and metaheuristics. CRC Press; doi:10.1201/9781420010749.
17. Kramer, O. (2017). Genetic Algorithm Essentials. Springer. 10.1007/978-3-319-52156-5.
18. Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. Science, 220(4598):671–680.
19. Glover, F., & Laguna, M. (2013). Tabu Search. In Handbook of Combinatorial Optimization (pp. 3261–3362). New York: Springer.
20. Rego, C., & Alidaee, B. (Eds.). (2006). Metaheuristic optimization via memory and evolution: tabu search and scatter search. Springer Science & Business Media.
21. Matthews, R. A. (1993), The Use of Genetic Algorithms in Cryptanalysis, Cryptologia, vol. 17, no. 2,pp. 187-201.
22. Spillman, R., Janssen, M., Nelson, B., & Kepner, M. (1993), Use of a genetic algorithm in the cryptanalysis of simple substitution ciphers. Cryptologia, 17(1), 31-44.
23. Clark, A. (1998), Optimisation Heuristics for Cryptology, Doctoral Dissertation, Queensland University of Technology, Australia.
24. Dimovski, A., & Gligoroski, D. (2003). Attack on the polyalphabetic substitution cipher using a parallel genetic algorithm. Swiss-Macedonian scientific cooperation trought SCOPES project.
25. Garg, P., & Sherry, A. M. (2005). Genetic algorithm & Tabu search attack on the mono-aiphanetic substitution cipher. Paradigm, 9(1), 106-109.
26. Verma, A. K., Dave, M., & Joshi, R. C. (2007). Genetic algorithm and tabu search attack on the mono-alphabetic substitution cipher i adhoc networks. In Journal of Computer science.
27. Omran, S. S., Al-Khalid, A. S., & Al-Saady, D. M. (2010). Using Genetic Algorithm to break a mono-alphabetic substitution cipher. In 2010 IEEE Conference on Open Systems (ICOS 2010) (pp. 63-67). IEEE.
28. Mudgal, P. K., Purohit, R., Sharma, R., & Jangir, M. K. (2017). Application of genetic algorithm in cryptanalysis of mono-alphabetic substitution cipher. In 2017 International Conference on Computing, Communication and Automation (ICCCA) (pp. 400-405). IEEE.

29. Garici, M. A., & Drias, H. (2005), Cryptanalysis of substitution ciphers using scatter Search. In LNCS Proceedings of International Work-Conference on the Interplay between Natural and Artificial Computation 2005, LNCS Springer Heidelberg, pp. 31-40.
30. Forsyth, W. S., & Safavi-Naini, R. (1993), Automated cryptanalysis of substitution ciphers, Cryptologia, 17 (4), 407-418.
31. Toemeh, R., & Arumugam, S. (2007). Breaking transposition cipher with genetic algorithm. Elektronika ir Elektrotechnika, 79(7), 75-78.
32. Song, J., Yang, F., Wang, M., & Zhang, H. (2008). Cryptanalysis of transposition cipher using simulated annealing genetic algorithm. In International Symposium on Intelligence Computation and Applications (pp. 795-802). Springer, Berlin, Heidelberg.
33. Muhajjar, R. A. (2010). Use of genetic algorithm in the cryptanalysis of transposition ciphers. basrah journal of science, 28(1A english), 49-57.
34. Al-Khalid, A. S., Omran, S. S., & Hammood, D. A. (2013). Using genetic algorithms to break a simple transposition cipher. In 6th International Conference on Information Technology ICIT.
35. Garg, P. (2009). Genetic algorithms, tabu search, and simulated annealing: a comparison between three approaches for the cryptanalysis of transposition cipher. Journal of Theoretical & Applied Information Technology, 5(4).
36. Mishra, G., & Kaur, S. (2015). Cryptanalysis of transposition cipher using hill climbing and simulated annealing. In Proceedings of Fourth International Conference on Soft Computing for Problem Solving (pp. 293-302). Springer, New Delhi.
37. Spillman, R. (1993). Cryptanalysis of knapsack ciphers using genetic algorithms. Cryptologia, 17(4), 367-377.
38. Yaseen, I. F., & Sahasrabuddhe, H. V. (1999). A genetic algorithm for the cryptanalysis of Chor-Rivest knapsack public key cryptosystem (PKC). In Proceedings Third International Conference on Computational Intelligence and Multimedia Applications. ICCIMA'99 (Cat. No. PR00300) (pp. 81-85). IEEE.
39. Garg, P., Shastri, A., & Agarwal, D. C. (2007). An enhanced cryptanalytic attack on Knapsack Cipher using Genetic Algorithm. International Journal of Computer and Information Engineering, 1(12), 4071-4074.
40. Ramani, G., & Balasubramanian, L. (2011). Genetic algorithm solution for cryptanalysis of knapsack cipher with knapsack sequence of size 16. International Journal of Computer Applications, 35(11), 17-23.
41. Song, J., Zhang, H., Meng, Q., & Wang, Z. (2007). Cryptanalysis of four-round DES based on genetic algorithm. In 2007 International Conference on Wireless Communications, Networking and Mobile Computing (pp. 2326-2329). IEEE.
42. Vimalathithan, R., & Valarmathi, M. L. (2009). Cryptanalysis of S-DES using genetic algorithm. International Journal of Recent Trends in Engineering, 2(4), 76.
43. Sathya, S. S., Chithralekha, T., & Anandakumar, P. (2010). Nomadic Genetic Algorithm for Cryptanalysis of DES 16. International Journal of Computer Theory and Engineering, 2(3), 1793-8201.
44. Sharma, L., Pathak, B. K., & Sharma, R. G. (2012). Breaking of simplified data encryption standard using genetic algorithm. Global Journal of Computer Science and Technology.
45. Al Adwan, F., Al Shraideh, M., & Al Saidat, M. S. (2015). A genetic algorithm approach for breaking of simplified data encryption standard. International Journal of Security and Its Applications, 9(9), 295-304.

46. Dworak, K., & Boryczka, U. (2017). Genetic algorithm as optimization tool for differential cryptanalysis of DES6. In International Conference on Computational Collective Intelligence (pp. 107-116). Springer, Cham.
47. Nalini, N., & Rao, G. R. (2005). Cryptanalysis of simplified data encryption standard via optimization heuristics. In 2005 3rd International Conference on Intelligent Sensing and Information Processing (pp. 74-79). IEEE.
48. Nalini, N. (2006). Cryptanalysis of block ciphers via improved simulated annealing technique. In 9th International Conference on Information Technology (ICIT'06) (pp. 182-185). IEEE.
49. Soyjaudah, K. M. S. (2012). Cryptanalysis of simplified-data encryption standard using tabu search method. In International Conference on Information Processing (pp. 561-568). Springer, Berlin, Heidelberg.
50. Cowan, M. J. (2008). Breaking short playfair ciphers with the simulated annealing algorithm. Cryptologia, 32(1), 71-83.
51. Clark, J. A., Jacob, J. L., & Stepney, S. (2005). The design of S-boxes by simulated annealing. New Generation Computing, 23(3), 219-231.