A Review on Swarm Intelligence Techniques in Automated Cryptanalysis of Classical Substitution Cipher

A Review on Swarm Intelligence Techniques in Automated Cryptanalysis of Classical Substitution Cipher

Ashish Jain[1], Santosh Kumar Vishwakarma[1], Prakash Chandra Sharma[1] and Nirmal Kumar Gupta[1]

[1]School of Computing and Information Technology, Manipal University Jaipur, Jaipur-Ajmer Express Highway, Dehmi Kalan, Near GVK Toll Plaza, Jaipur, Rajasthan 303007

E-mail: santosh.kumar@jaipur.manipal.edu

Abstract. Between the year 2006 and 2019, a considerable new and different swarm intelligence techniques have been presented in the literature for automated cryptanalysis of classical substitution cipher. This paper compares the performance of these new and different swarm intelligence techniques. Three main comparison measures are considered to assess the performance of presented swarm intelligence techniques: efficiency, effectiveness, and success rate. To the best of author knowledge first time this kind of review has been carried out. It is noteworthy that among the presented swarm intelligence techniques the performance of cuckoo search technique is best with respect to all the measures.

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 are utilized to determine an adequate solution [2].

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]. In this paper the cryptanalysis problems that are considered to solve are comes in the class of NP-complete problems.

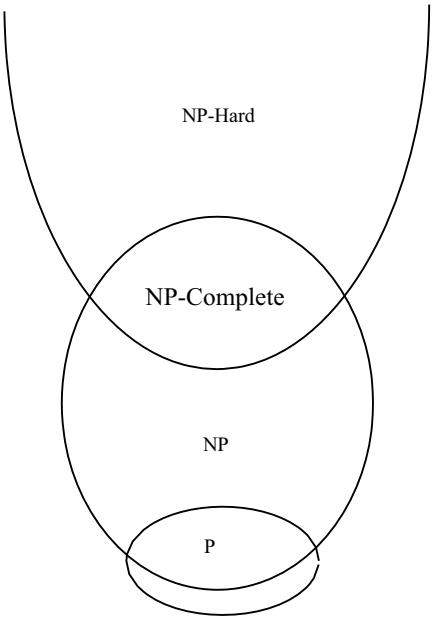

Figure 1. P, NP, NP-complete and NP-hard [1]

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 comes in the class of NP-complete problem [3][4][5][6]. 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 [7][8][9]. However, automated attacks can be formed using swarm intelligence techniques that can search the ciphers key in an acceptable amount of time [10][11]. The automated cryptanalysis of the classical substitution cipher is considered in this paper. For details about this cipher the reader can refer [9].

2. Performance Measurement Criteria

The substitution 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 substitution cipher does not altered the character frequency distribution significantly. Therefore, the swarm intelligence techniques 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 swarm intelligence 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 swarm intelligence techniques in the result section.

3. Literature Review

The application of swarm intelligence techniques in automated attacks of classical substitution ciphers was first reported in 2006 in [12][13], and the outcomes have demonstrated that swarm intelligence strategies are exceptionally efficient and effective. With this inspiration, numerous swarm intelligence techniques have been reported for mounting automated attacks on the substitution cipher, for example, particle swarm optimization, bees algorithm, ant colony optimization, firefly algorithm, and cuckoo search. Among all these algorithms the cuckoo search proposed by Jain and Chaudhary [9][11] has shown the best performance with respect to all the comparison criteria. For automated cryptanalysis of the "classical substitution cipher, hereinafter, substitution cipher" multiple swarm intelligence techniques have been used in the past that have been mentioned above. Below we describe the standard form of these techniques in brief.

Kennedy and Eberhart [14] proposed a population-based swarm intelligence strategy, namely, particle swarm optimization. This strategy has been formulated by means of reproduction investigations of winged creatures rushing. This strategy is starts with a random population of individuals called "particles". With each particle (or molecule) following parameters are associated: position and velocity. During fly in the multidimensional search space, every molecule changes its position dependent on its own understanding and of neighboring particles. That is, the particle tries to reach to the optimal solution by using its best position and the neighboring best position. The closeness of every molecule to the global optimum is assessed using a fitness function. For point-by-point depiction on the particle swarm optimization the reader can refer [15].

Pham et al. [16] have proposed a population-based swarm intelligence strategy, namely, bees algorithm. The algorithm copies the nourishment rummaging conduct of swarms of bumble bees. In its essential form, the algorithm plays out a sort neighborhood search joined with arbitrary hunt. For point- by-point depiction on the bees algorithm the reader can refer [16].

Bilchev and Parmee [17] developed the first ant colony optimization technique for continuous function optimization. Local optimization was focused on this method. The local search strategy was extended to a global search strategy by Wodrich and Bilchev [18] which was further modified by Jayaraman et al. [19]. This approach performs a bilevel search, with a local search component to exploit good regions of the search space, and a global search component to explore bad regions [19]. For point-by-point depiction on the ant colony optimization the reader can refer [15].

Yang [20] has proposed a population-based swarm intelligence strategy, namely, firefly algorithm. This algorithm is initialized with a random population of individuals called “fireflies”. This technique has been devised based on the flashing pattern of tropical fireflies. Flashing pattern can be idealized using three rules which are mentioned in [20]. For point-by-point depiction on the firefly algorithm the reader can refer [20].

Yang and Deb [21] has proposed a population-based swarm intelligence strategy, namely, cuckoo search. This method is starts with a random population of individuals called “nests”. A best nest (a nest with optimal value w. r. t. objective function) is picked from the available nests. Afterwards, from existing nests one more nest is picked randomly, say, ith nest. Using ith nest and best nest, a new nest is generated via Lévy flights. In this process a small fraction of worst nests is also abandoned by new nests. After numerous repetitions, the process stops, and we get a solution with good value regarding the objective function. For point-by-point depiction on the cuckoo search the reader can refer [21][22].

In the literature, particle swarm optimization (PSO), bees algorithm, ant colony optimization, firefly algorithm, and cuckoo 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 1.

Table 1. Applications of Cryptology Problems Solved using PSO, Bees Algorithm, Ant Colony Optimization, Firefly Algorithm, and Cuckoo Search

Authors [Reference]	**Swarm Intelligence Technique Used**	**Problem Solved**	**Application**
Uddin and Youssef [12] Ali et al. [23] Sadiq [24]	PSO	Automated Cryptanalysis of Classical Substitution Cipher	Modern Substitution Ciphers Uses Functions of Classical Substitution Cipher in a Complicated Way [8].
Ali [25]	Bees Algorithm		
Uddin and Youssef [13] Grari et al. [26]	Ant Colony Optimization		
Luthra et al. [27] Singh et al. [28]	Firefly Algorithm		
Jain and Chaudhari [9] Jain and Chaudhari [11]	Cuckoo Search		
Hameed and Hmood [29] Jassim [30]	PSO	Automated Cryptanalysis of Classical Transposition Cipher	Modern Transposition Ciphers Uses Functions of Classical Transposition Cipher in a Complicated Way [8].
Russell et al. [31] Mekhaznia and Menai [32] Heydari and Senejani [33]	Ant Colony Optimization Cuckoo Search		
Shahzad et al. [34] Abd-Elmonim et al. [35] Pandey and Mishra [36] Jadon et al. [37]	PSO	Automated Cryptanalysis of Data Encryption	DES is a Modern Block Cipher Used for Encryption of Confidential Information [8].

Khan et al. [38] Grari et al. [39]	Ant Colony Optimization	Standard (DES)	
Amic et al. [40]	Firefly Algorithm		
AbdulHalim [41]	PSO	Automated Cryptanalysis of Knapsack Cipher	Knapsack Cipher is a Reasonable Alternative, Particularly for Security of Little Implanted Gadgets, e.g., Cellular Devices [8].
Palit et al. [42]	Firefly Algorithm		
Bhateja et al. [10] Bhateja et al. [43]	Cuckoo Search PSO	Automated Cryptanalysis of Vigenere Cipher	Modern Substitution Ciphers Uses Functions of Vigenere Substitution Cipher in a Complicated Way [8].
Grari et al. [44]	Ant Colony Optimization	Automated Cryptanalysis of Simplified Advanced Encryption Standard (AES)	AES is a Modern Block Cipher Used for Encryption of Confidential Information [8].
Ahmed et al. [45]	Firefly Algorithm	Construction of Novel Substitution- Box	Novel Substitution-Box Can be Used in Design of Stream Ciphers. Stream Ciphers are Used for Encryption of Confidential Information [45]

4. Comparative Analysis

Recall from Section 2, the standard strategy for escalating attacks on the substitution 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 swarm intelligence 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. Thusly, the fitness of *K* is determined. Formally, Eq. (1) is utilized for statistics comparison.

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

For clarification on Eq. (1) the reader can refer [9]. In any case, in the writing it has demonstrated that typically the best operational reason for a fitness function utilized in automated cryptanalysis of substitution ciphers are the bigrams only, i.e., n = 2 [9, 11-13, 26]. These realities persuade us to utilize

the fitness function which is mentioned in Eq. (2) that depends on just the bigrams.

$$Cost_k = \sum_{i,j\in\zeta} |k^{\hat{b}}_{i,j} - d^{\hat{b}}_{i,j}| \quad (2)$$

Experiment. For performing experiments, the considered swarm intelligence techniques have been implemented in Java. We followed the guidelines reported in the respective papers during implementation of each of the swarm intelligence techniques. Known ciphertext, length of the ciphertext, and the English language bigram statistics are input to every algorithm as shown in figure 2 and table 2.

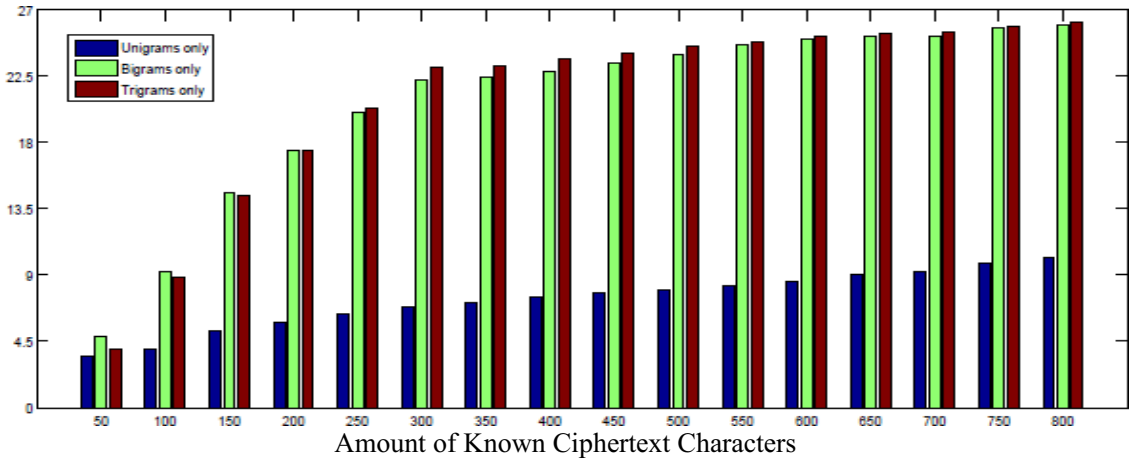

Figure 2. Plotting of Average Outcomes: Number of Key Components Accurately Recovered Utilizing Cost Function Dependent on Unigrams Just, Bigrams Just and Trigrams as it were

Table 2. Cryptanalysis Results Obtained Through Various Swarm Intelligence Strategies on the Substitution Cipher

Year	Authors [Reference]	Swarm Intelligence Techniques Used	Maximum Number of Ciphertext Characters Used	Average Number of Key Elements Correctly Recovered out of 27	Mean Performance Time (in seconds) to recover the key
2006	Uddin and Youssef [12]	PSO	1000	24.87	0.437
2006	Uddin and Youssef [13]	Ant Colony Optimization	1000	25.02	0.367

2010	**Ali et al. [23]**	**PSO**	**1000**	**24.93**	**0.441**
2011	Luthra et al. [27]	Firefly Algorithm	1000	24.61	0.320
2012	Sadiq [24]	PSO	1000	24.95	0.435
2013	Singh et al. [28]	Firefly Algorithm	1000	24.73	0.320
2013	Ali [25]	Bees Algorithm	1000	24.13	0.523
2016	Grari et al. [39]	Ant Colony Optimization	1000	25.13	0.357
2015, 2019	Jain and Chaudhari [9, 11]	Cuckoo Search	**800**	**26.17**	**0.137**

Analysis of Results. Regarding all the performance criteria, we mention the obtained results in the Table 2. Note that the swarm intelligence technique that takes a greater number of ciphertext characters are said to be less effective than the swarm intelligence technique which takes lesser number of ciphertext characters. From the obtained results, we can observe that the cuckoo search proposed in [9] and [11] is most effective because taking only 800 number of ciphertext characters for successful recovery of key. From the obtained results, we can clearly observe that the cuckoo search technique proposed in [9] and [11] takes only 800 ciphertext characters and as an outcome able to recover 26.17 number of key elements. The time taken by the algorithm is also lowest which is 0.137 seconds.

5. Conclusions

The efficient, effective, and successful utilization of various swarm intelligence techniques in solving the substitution cipher is presented. The following outcomes are noted: (1) In terms of successful attacks and efficiency the PSO proposed by Sadiq [24] performs better than the PSO proposed by Uddin and Youssef [12] and Ali et al. [23]. (2) In terms of successful attacks and efficiency the ant colony optimization proposed by Grari et al. [39] performs better than the ant colony optimization proposed by Uddin and Youssef [13]. (3) In terms of successful attacks, the firefly algorithm proposed by Singh et al. [28] performs better than the firefly algorithm proposed by Luthra et al. [27]. (4) In terms of all the measures the performance of Bees algorithm proposed by Ali [25] is worst. (5) In terms of all the measures the performance of the cuckoo search technique proposed by Jain and Chaudhari [9, 11] is extremely significantly best.

6. 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), 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] Jain, A. and Chaudhari, N. S. (2015), A New Heuristic Based on the Cuckoo Search for Cryptanalysis of Substitution Ciphers. *In LNCS Proceedings of the 21st International Conference on Neural Information Processing* 2015, Istanbul, Turkey, LNCS Springer Cham, pp. 206-215.
[12] Uddin, M. F., & Youssef, A. M. (2006). Cryptanalysis of simple substitution ciphers using particle swarm optimization. *In 2006 IEEE International Conference on Evolutionary Computation* (pp. 677-680). IEEE.
[13] Uddin, M. F., & Youssef, A. M. (2006). An artificial life technique for the cryptanalysis of simple substitution ciphers. *In 2006 Canadian Conference on Electrical and Computer Engineering* (pp. 1582-1585). IEEE.
[14] Kennedy J, Eberhart RC et al. (1995) Particle swarm optimization. *In: IEEE international conference on neural networks*, vol 4, pp 1942–1948, IEEE.
[15] Engelbrecht, A. P. (2007). Computational intelligence: an introduction. *John Wiley & Sons*.
[16] Pham, D. T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., & Zaidi, M. (2005). The bees algorithm. *Technical Note, Manufacturing Engineering Centre*, Cardiff University, UK.
[17] Bilchev, G., & Parmee, I. C. (1995). The ant colony metaphor for searching continuous design spaces. In AISB workshop on evolutionary computing (pp. 25-39). *Springer, Berlin, Heidelberg*.
[18] Wodrich, M. (1997). Cooperative distributed search: The ants way. *Control and Cybernetics*, 26, 413-446.
[19] Jayaraman, V. K., Kulkarni, B. D., Karale, S., & Shelokar, P. (2000). Ant colony framework for optimal design and scheduling of batch plants. *Computers & Chemical Engineering*, 24(8), 1901-1912.
[20] Yang, X. S. (2008). Firefly algorithm. *Nature-inspired metaheuristic algorithms*, 20, 79-90.
[21] Yang, X. S., & Deb, S. (2009). Cuckoo search via Lévy flights. *In 2009 World congress on nature & biologically inspired computing (NaBIC)* (pp. 210-214). IEEE.
[22] Yang, X. S. (2014). Nature-inspired optimization algorithms. *Elsevier*.
[23] Ali, I. K., Sadiq, A. T., & Salih, H. H. (2010). Attack on the Simple Substitution Ciphers Using Particle Swarm Optimization. *Engineering and Technology Journal*, 28(11), 2151-2161.
[24] Sadiq, A. T. (2012). Mutation-based particle swarm optimization (MPSO) to attack classical cryptography methods. *Journal of Advanced Computer Science and Technology Research*, 2, 50-65.
[25] Ali, I. K. (2013). Cryptanalysis of simple substitution ciphers using bees algorithm. *Journal of*

Baghdad College of Economic sciences University, (36), 373-382.

[26] Grari, H., Azouaoui, A., & Zine-Dine, K. (2016). A novel ant colony optimization based cryptanalysis of substitution cipher. *In International Afro-European Conference for Industrial Advancement* (pp. 180-187). Springer, Cham.

[27] Luthra, J., & Pal, S. K. (2011). A hybrid firefly algorithm using genetic operators for the cryptanalysis of a monoalphabetic substitution cipher. *In 2011 World congress on information and communication technologies* (pp. 202-206). IEEE.

[28] Singh, A. P., Pal, S. K., & Bhatia, M. P. S. (2013). The firefly algorithm and application in cryptanalysis of monoalphabetic substitution ciphers. *American Journal of Computer Science and Engineering Survey*, 1(1), 33-52.

[29] Hameed, S. M., & Hmood, D. N. (2010). Particles swarm optimization for the cryptanalysis of transposition cipher. *Al-Nahrain Journal of Science*, 13(4), 211-215.

[30] Jassim, M. K. (2017). Improved PSO algorithm to attack transposition cipher. *Engineering and Technology Journal*, 35(2 Part (B) Scientific), 144-149.

[31] Russell, M. D., Clark, J. A., & Stepney, S. (2003). Making the most of two heuristics: Breaking transposition ciphers with ants. *In The 2003 Congress on Evolutionary Computation*, 2003. CEC'03. (Vol. 4, pp. 2653-2658). IEEE.

[32] Mekhaznia, T., & Menai, M. E. B. (2014). Cryptanalysis of classical ciphers with ant algorithms. *International Journal of Metaheuristics*, 3(3), 175-198.

[33] Heydari, M., & Senejani, M. N. (2014). Automated cryptanalysis of transposition ciphers using cuckoo search algorithm. *International Journal of Computer Science and Mobile Computing*, *3*(1), 140-149.

[34] Shahzad, W., Siddiqui, A. B., & Khan, F. A. (2009). Cryptanalysis of four-rounded DES using binary particle swarm optimization. *In Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers* (pp. 2161-2166).

[35] Abd-Elmonim, W. G., Ghali, N. I., Hassanien, A. E., & Abraham, A. (2011). Known-plaintext attack of DES-16 using Particle Swarm Optimization. *In 2011 Third World Congress on Nature and Biologically Inspired Computing* (pp. 12-16). IEEE.

[36] Pandey, S., & Mishra, M. (2012). Particle swarm optimization in cryptanalysis of DES. *Int. J. of Advanced Research in Computer Engineering & Technology*, 1(4), 379-381.

[37] Jadon, S. S., Sharma, H., Kumar, E., & Bansal, J. C. (2012). Application of binary particle swarm optimization in cryptanalysis of DES. *In Proceedings of the International Conference on Soft Computing for Problem Solving* (SocProS 2011) December 20-22, 2011 (pp. 1061-1071). Springer, India.

[38] Khan, S., Shahzad, W., & Khan, F. A. (2010). Cryptanalysis of four-rounded DES using ant colony optimization. *In 2010 International Conference on Information Science and Applications* (pp. 1-7). IEEE.

[39] Grari, H., Azouaoui, A., & Zine-Dine, K. (2018). Ant Colony Optimization for Cryptanalysis of Simplified-DES. *In International Conference on Advanced Intelligent Systems for Sustainable Development* (pp. 111-121). Springer, Cham.

[40] Amic, S., Soyjaudah, K. S., Mohabeer, H., & Ramsawock, G. (2016). Cryptanalysis of DES-16 using binary firefly algorithm. *In 2016 IEEE International Conference on Emerging Technologies and Innovative Business Practices for the Transformation of Societies* (EmergiTech) (pp. 94-99). IEEE.

[41] AbdulHalim, M. F., Bara'a, A. A., & Hameed, S. M. (2008). A binary particle swarm optimization for attacking knapsacks cipher algorithm. *In 2008 International Conference on Computer and Communication Engineering* (pp. 77-81). IEEE.

[42] Palit, S., Sinha, S. N., Molla, M. A., Khanra, A., & Kule, M. (2011). A cryptanalytic attack on the knapsack cryptosystem using binary firefly algorithm. In 2011 *2nd International conference on computer and communication technology* (ICCCT-2011) (pp. 428-432). IEEE.
[43] Bhateja, A., Kumar, S., & Bhateja, A. K. (2013). Cryptanalysis of Vigenere cipher using particle swarm optimization with Markov chain random walk. *International Journal on Computer Science and Engineering*, 5(5), 422.
[44] Grari, H., Azouaoui, A., & Zine-Dine, K. (2019). A cryptanalytic attack of simplified-AES using ant colony optimization. *International Journal of Electrical & Computer Engineering* (2088-8708).
[45] Ahmed, H. A., Zolkipli, M. F., & Ahmad, M. (2019). A novel efficient substitution-box design based on firefly algorithm and discrete chaotic map. *Neural Computing and Applications*, 31(11), 7201-7210.