# A Novel Ant Colony Optimization Based Cryptanalysis of Substitution Cipher

Hicham Grari(✉), Ahmed Azouaoui, and Khalid Zine-Dine

FS, LAROSERI Laboratory, Chouaib Doukkali University, El Jadida, Morocco
grari.hicham@gmail.com,
{azouaoui.a,zinedine}@ucd.ac.ma

**Abstract.** In this paper, we present a novel Ant Colony Optimization (ACO) based attack for cryptanalysis of a simple substitution Cipher. A known only cipher text attack is used to recovering the key based on the letter frequency language, a new algorithm is introduced with a Fitness Function that have a good correlation with the number of key elements recovered. Our approach shows promising results when compared with other techniques.

**Keywords:** Cryptanalysis · ACO · Substitution cipher · Pheromone

## 1 Introduction

Recently, the need for security is constantly increasing, with the development and evolution of communications networks, especially the cryptology which has become a scientific discipline dealing with Confidentiality, Integrity and Authentication.

They are two fields in cryptology, cryptography and cryptanalysis. If Cryptography is the study of methods for sending messages in disguised forms called ciphertext, Cryptanalysis is the art of breaking ciphertext. Particularly, Cryptanalysis is the study of mathematical techniques to defeat cryptographic algorithms.

Actually, research in this area, are intended to use heuristic algorithms. It may appear an efficiency way to break complex ciphers. Ant Colony Optimization (ACO) [1] is a well-known meta-heuristics that were successfully used to produce approximate solutions for a large variety of optimization problems. In this paper we investigate the use of ACO in automated cryptanalysis of simple substitution cipher.

Peleg and Rosenfeld [2] was the first one to introduce the Artificial intelligence techniques in the field of cryptography to breaking substitution ciphers by modeling the problem as a probabilistic labeling problem, they demonstrated the application of relaxation methods to the solution of substitution ciphers.

Spillman et al. [3] presented an attack on simple substitution cipher using genetic algorithm. Bahler and King [4] used trigram statistics to calculate Trigram coefficients using Shannon's adjustment formula to converge towards the most probable key. Lucks [5] used an exhaustive search in a large on-line dictionary for words that satisfy constraints on word length, letter position and letter multiplicity. Additionally, two efficient attacks of the substitution cipher based on the genetic algorithm and tabu search are implemented by Clark [6].

A. Abraham et al. (eds.), *Proceedings of the Third International Afro-European Conference for Industrial Advancement—AECIA 2016*, Advances in Intelligent Systems and Computing 565, DOI 10.1007/978-3-319-60834-1_19

Ant Colony Optimization (ACO) was used successfully in breaking transposition ciphers [7] by M.D. Russell, J.A. Clark.

Recently, Ashish Jain and Narendra [8] proposed a new heuristic based on the cuckoo search for cryptanalysis of substitution ciphers. A comparison is done with many others heuristic like GA and Tabu search. Similarly, Uddin Mohammad Faisal [9] proposed Cryptanalysis of Simple Substitution Ciphers Using Ant colony Optimization. They showed that ACO provides a very powerful tool for the cryptanalysis of substitution ciphers using a ciphertext only attack.

Ant colony Optimization are also used for cryptanalysis of DES by Salabat Khan, A. Armughan and Mehr Y Durrani [10], and for the design of Substitution–Box by A. Musheer a,*, Deepanshu Bhatiaa, Yusuf Hassana [11].

The remainder of this paper is organized as follows. In the next Section, we introduce simple substitution cipher. In Sect. 3, we present the basic and background of Ant colony optimization meta-heuristic. The fully automated attack with a novel Fitness Function is given in Sect. 4, with experimental results in Sect. 5. Finally, conclusions are given in Sect. 6.

## 2 Simple Substitution Cipher

Basically, the idea of a simple substitution cipher is to replace each letter in the plaintext by a fixed other letter in order to get an encrypted text called ciphertext.

For instance, to encrypt the word "SIMPLE" using the key mentioned in Fig. 1 we have to replace 'S' by 'W', 'I' by 'M' … etc.

| | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Letters : | 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 |
| Key : | Z | K | L | C | J | G | P | X | M | N | I | V | U | D | Y | A | R | O | W | T | B | H | S | E | F | Q |

**Encryption**:

S ⟶ W
I ⟶ M
M ⟶ U
P ⟶ A
L ⟶ V
E ⟶ J

**Fig. 1.** Simple substitution cipher

Although the number of possible keys is very large ($26! \approx 2^{88}$ possibles key), this cipher is not very strong, the main disadvantage of this encryption algorithm is that the character frequency statistics (or n-grams that indicate the frequency distribution of all possible instances of 'n' adjacent characters frequency) in the encrypted text can be determined and compared to the standard frequency distribution of the natural language used.

## 3 Ant Colony Optimization

Optimization techniques have got a significant importance in determining efficient solutions of different complex and hard problems. In particular, Ant Colony Optimization [2] which represents a class of population-based meta-heuristics inspired by the behavior of real ant colonies. With their ability to find shortest paths between food sources and their nest, using pheromone communication.

Ants initially explore randomly the environment surrounding their nest, when an ant finds food; it walks backs to the colony leaving behind a pheromone trail that may depend on the quantity and the quality of the food. The other ants of the colony are expected to follow the path of greater pheromone trail with higher probability; old paths are less likely to be used because of the pheromone evaporation mechanism, allowing forgetting bad solutions. This simple idea is implemented by the ACO methods to resolve and address hard optimization problems.

## 4 Proposed Algorithm

As mentioned above, simple substitution cipher does not resist to the frequency analyse, several works [6, 8, 9] take advantage of this weakness to attacks this encryption algorithm. In our work, analyses were done by comparing frequency of single character (unigram) and double character (bigram) of the ciphertext with the reference language.

The general formula used in previous works [6, 8, 9] to measure the quality of a guessed key is shown in Eq. (1).

$$\omega_1 \times \sum\nolimits_{i \in A} K^u(i) - R^u(i) + \omega_2 \times \sum\nolimits_{i \in A} K^u(i,j) - R^u(i,j) \tag{1}$$

With $R^u(i)$, $R^u(i,j)$ and $K^u(i)$, $K^u(i,j)$ are the reference language unigram and bigram statistics and decrypted message unigram and bigram statistics.

With $\omega_1$ and $\omega_2$ are coefficients of unigram and bigram.

And A is the set of Alphabet $A = \{A, B, C, \ldots, Z\}$.

To apply the ACO meta-heuristic to our problem, a first step is to map the problem on a construction graph. Given a fully connected directed graph of 26 nodes, corresponding to 26 characters of Alphabet. A key is represented by a Hamiltonian path constructed by an ant moving across the graph.

At each construction step, ant ‘k’ applies a probabilistic action choice rule, to decide which node to visit next. In particular, the probability with which ant k, currently at node $i$, chooses to move to node $j$ is given by Eq. (2).

$$P(i,j) = \begin{cases} 0 & \textit{if already visited} \\ \frac{\tau(i,j)^{\alpha} \rho(i,j)^{\beta}}{\sum_{i \in S} \tau(i,j)^{\alpha} \rho(i,j)^{\beta}} & \textit{Otherwise} \end{cases} \tag{2}$$

Two parameters are used to calculate the probability of moving to a particular node; first, the amount of pheromone $\tau(i, j)$ on the connecting edge. And second the heuristic value $\rho(i,j)$ representing the a priori knowledge of desirability of the choice.

In our approach the heuristic value is typically inversely proportional to the distance between the unigram frequency of the reference language statistics and the target test key, given by Eq. (3).

$$\rho(i,j) = \frac{1}{|K^u(i) - D^u(i)|} \tag{3}$$

The parameters α and β are influencing factors of pheromone and heuristic value, respectively.

The solution construction terminates once all node have been visited, each complete tour of an ant from a source node to a destination node represent a solution to the problem at hand. The evaluation of a solution is done using a fitness function presented in Eq. (1).

## 4.1 Fitness Function

In the evolution cryptanalysis algorithm, evaluation methods or the design of fitness function, it is another influence cryptanalysis of one of the difficult points of success or failure. In this paper, the fitness function proposed in Eq. (4) is studied.

$$Cost(K) = \omega_1 \times \sum\nolimits_{i \in A} \frac{K^u(i) - D^u(i)}{M^u(i)} + \omega_2 \times \sum\nolimits_{i \in A} \frac{K^u(i,j) - D^u(i,j)}{M^u(i,j)} \tag{4}$$

With $M^u(i) = \frac{K^u(i) + D^u(i)}{2}$ and $M^u(i,j) = \frac{K^u(i,j) + D^u(i,j)}{2}$

In order to reduce the effect of the characters having a larger percentage of appearance versus those who have a low percentage (Example: The statistic of the letter “e” is 12.5% but “z” have only 0.01).

## 4.2 Pheromone Update and Evaporation

After all the ants have constructed their tours, the pheromone trails are updated by adding pheromone on the arcs the ants have crossed in their tours using Eq. 5.

$$\tau_{i,j} = \tau_{i,j} + \frac{C}{Cost(K)} \tag{5}$$

Where ‘C’ is some constant to be determined.

The pheromone updates depend on the solution quality, smaller the value of Cost (K) (good solution) higher is the pheromone value added to the previous pheromone value on an edge.

In real ant colonies, pheromone intensity decreases over time because of evaporation mechanism. In ACO evaporation is simulated by applying an appropriately defined pheromone evaporation rule as shown in Eq. 6. Pheromone evaporation reduces the influence of the pheromones deposited in the early stages of the search, this also favoring the forgetting of poor choices done in the past.

$$\tau_{i,j} = \tau_{i,j} \times \sigma\{whith\ \sigma\ will\ be\ between\ 0\ et\ 1\} \tag{6}$$

### 4.3 Proposed Algorithm

```
1. Initialize the pheromone trails.
2. Generate a solution based on probability equation (2).
3. Evaluate Solution according to equation (4).
4. Update the best ant solution.
5. Update pheromone value and evaporation using equation (5) and (6).
6. Repeat this process N times (Number of iteration).
7. Repeat the steps from 2 - 6 until a maximum number of run (R) have been attained or threshold of Fitness Function is reached.
```

## 5 Experimental Results and Discussions

The parameters such as α, β (weight of pheromone and heuristic value), N (Number of Iteration), R (Number of Run), 'C' and $\sigma$. were fine-tuned by a combination of several experiments in order to optimize the cryptanalysis process. We have implemented our algorithm with c++ language.

First, we investigated the case when $\omega_2 = 0$ in Eq. (4) (Fitness Function based on unigram statistics only).

In Order to assess the relevance of our Fitness Function, the correlation coefficient 'ρ' between the cost function (4) and the number of corrected key elements is calculated, for different Amount of known ciphertext, (based on 1000 random Key). This coefficient is used to evaluate the relationship between the cost function and the number of recovered key elements.

In the case of a perfect direct (increasing) linear relationship we have ρ = 1, its means that we have a good mechanism for evaluation of generated key, therefore the convergence of our algorithm to the best key is most guaranteed. Inversely, in a perfect decreasing linear relationship we have ρ = −1, and some value in the open interval (−1, 1), indicating the degree of linear dependence between the variables. As it approaches zero there is less of a relationship. Figure 2 show the value of the correlation coefficient (displayed in absolute value) of the Fitness function used in our approach (4) named F1, and that used in literature (1) named F2. Graph shows that the correlation coefficient in proposed method is better in all the cases of amount of known ciphertext.

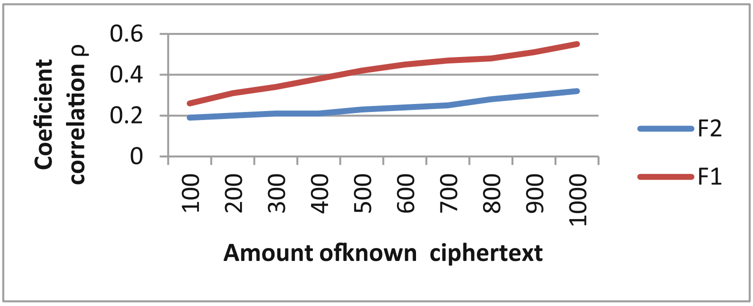

**Fig. 2.** Correlation coefficient value of fitness functions F1 and F2

In Fig. 3, the comparison between our Fitness Function and that used previously, the X-axis shows the amount of known ciphertext and Y-axis shows the Number of key elements correct.

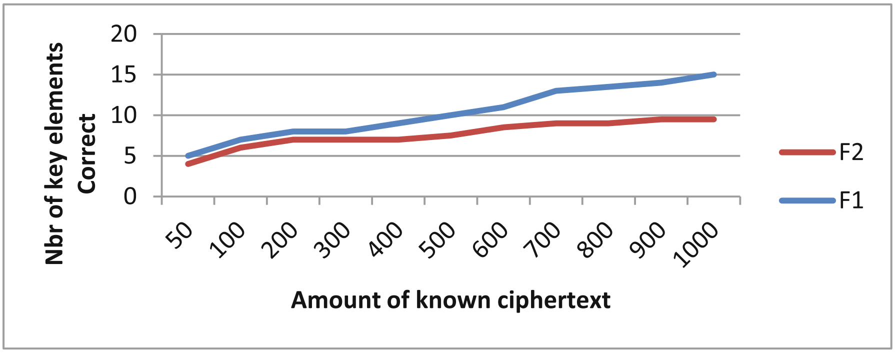

**Fig. 3.** Number of key elements correct of fitness functions F1 and F2.

Comparison results show that in our approach the number of key elements recovered is higher for different Amount of known cipher text. Particularly, with significant amount of known cipher text.

In the second part, we investigate the case when w1 = 0 in Eq. (4) (Fitness Function based on bigram statistics only).

The comparison between our Fitness Function and that used previously is presented in Fig. 4. In this case; the difference is less significant than the first case (Fitness Function based on unigram only) for different amount of cipher text.

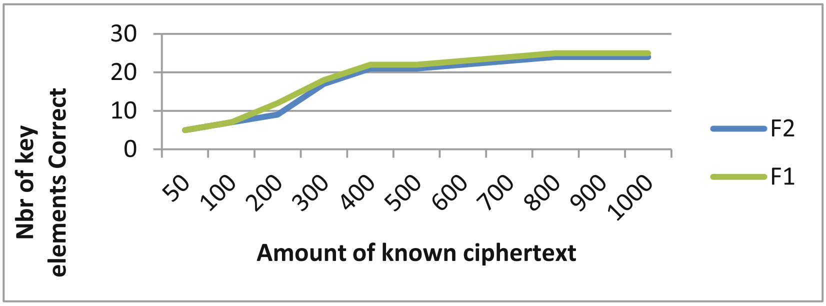

**Fig. 4.** Number of key elements correct of fitness functions F1 and F2.

## 6 Conclusion

This paper demonstrates that the success of meta-heuristic based algorithms is dependent upon the availability of a suitable solution evaluation mechanism. Such a mechanism (a fitness function or a cost function) must accurately assess every feasible solution giving an indication of its optimality. There are several important avenues for future research, it will be a good idea to try others statistical distance (i.e., KullBack-Leibler divergence, Hellinger distance) to calculate the fitness function.

ACO provides a very powerful tool for the cryptanalysis of simple substitution ciphers, it is interesting to be applied to cryptanalysis of some others strong encryption algorithms like Data encryption Algorithm or AES (Advanced Encryption Standard).

## References

1. Dorigo, M., Maniezzo, V., Colorni, A.: The ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern. B **26**(2), 29–41 (1996)
2. Peleg, S., Rosenfeld, A.: Breaking substitution ciphers using a relaxation algorithm. Commun. ACM **22**(11), 598–605 (1979)
3. Spillman, R., Janssen, M., Nelson, B., Kepner, M.: Use of a genetic algorithm in the cryptanalysis of simple substitution ciphers. Cryptologia **17**(1), 31–44 (1993)
4. Bahler, D.R., King, J.C.: An implementation of probabilistic relaxation in the cryptanalysis of simple substitution systems. Cryptologia **16**(3), 219–225 (1992)
5. Lucks, M.: A constraint satisfaction algorithm for the automated decryption of simple substitution Ciphers. In: Proceedings of CRYPTO 1988, pp. 132–144 (1988)
6. Clark, A.: Optimisation heuristics for cryptology, PhD thesis, Queensland University of Technology (1998)
7. Russell, M.D., Clark J.A., Stepney, S.: Making the most of two heuristics: breaking transposition ciphers with ants. In: The Congress on Evolutionary Computation (CEC 2003), vol. 4, pp. 2653–2658 (2003)
8. Jain, A., Narendra, S.: A New Heuristic based on the cuckoo search for cryptanalysis of substitution ciphers. In: 22nd International Conference on ICONIP 2015, Istanbul, Turkey, 9–12 November (2015)

9. Uddin, M.F., Youssef, A.M.: An artificial life technique for the cryptanalysis of simple substitution ciphers electrical and computer engineering (2006)
10. Khan, S., Ali, A., Durrani, M.Y.: Ant-crypto, a cryptographer for data encryption standard. IJCSI **10**(1), 1694–1784 (2013)
11. Ahmad, M., Bhatiaa, D., Hassana, Y.: Novel Ant Colony Optimization based scheme for substitution box design. Procedia Comput. Sci. **57**, 570–580 (2015)