Use of a new approach to automated break transposition cipher system

IOP Conf. Series: Materials Science and Engineering **518** (2019) 052020 doi:10.1088/1757-899X/518/5/052020

Use of a new approach to automated break transposition cipher system

Ahmed Kareem Shibeeb[1] and Mohammed Hussein Ahmed[2]
[1]Department of Computer Systems, Technical Institute – Suwaira, Middle Technical University, Baghdad, Iraq.
[2]Department of Computer Science, College of Education, Al-Mustansiriyah University, Baghdad, Iraq.
Email: Ahmed.kareem@mtu.edu.iq, mohammedalbawi@uomustansiriyah.edu.iq

Abstract. Particle swarm optimization (PSO) based cryptanalysis has acquired much attention because it has a fast convergence rate. This paper investigates the use of a new approach which is PSO depending on the clustering algorithm (CLPSO) to break the transposition cipher system in appropriate time. The dynamic of CLPSO is different from existing PSO in population size, topology and the ways to find the best solution. CLPSO start's with using high population size (particles) and then applying preprocessing operation in order to reduce this population and then partition this population into several clusters based on using semi CLARANS algorithm and allow particles to share information in different clusters and give the particles ability to move from one cluster to another depending on the values of fitness function. Experimental results of the proposed CLPSO were very promising and the results proved that the CLPSO algorithm reduces the number of tries that needed to attack long key transposition cipher in almost real time using ciphertext-only attack. This new method allows recovering key length up to 35 with no more than 58.3 second as maximum consuming time. In this case study, different parameters such as: population sizes (100-5000), key size (10 - 35) and length of ciphertext (250 - 5000) were used.

1. Introduction

The main problem of cryptography systems is the trustworthy improvement of cryptography system where the main matter in cryptanalysis is how to find new practical manner to decipher the present system. The main goal of cryptography protects information by making them available only to authorize persons. In another hand, cryptanalysis is the process of retrieving the cleartext from existing ciphertext. It also can be defined as the ways to decrypt cryptography systems without knowing encryption key. Cryptanalyst presumes that the assailants know the cryptosystem, but don't know the key or algorithm. In many encryption systems, cryptanalysis uses to check the strength of cryptosystem [1]. Particle swarm optimization algorithm is a powerful manner to achieve cryptanalysis on different cryptographic systems [2]. Data mining considers as useful nuggets of information amidst massive amount of data and data clustering is a part of the data mining which it's the process of grouping data objects into a number of clusters. This paper focused on using of Particle Swarm Optimization based on semi partition clustering algorithm (CLPSO) to cryptanalysis columnar transposition cipher system.

1.1. Related work

There are many previous ways for cryptanalysis transposition cipher system. This paper reviews some of these in this section:

- Giddy and Naini in [3]. This paper used simulated annealing (SA) to break columnar transposition ciphers and used bigram analysis with random transformations called G-Transpositions to determine the fitness value. This work was successful to recover about 80% elements from the key length 15,20 and 25.
- Dimovski and Gligoroski in [4]. In this work, multiple optimization techniques were used to attack columnar transposition cipher. These techniques were a genetic algorithm, simulated annealing, and tabu search. The fitness values were calculated based on bigram statistics only. This research implemented of all three techniques to decrypt transposition cipher with different length and the key size up to 30. Experimental results showed that the three techniques can recover 12 parts from the key length 15, recover 17 parts from the key length 20 and recover 25 letters from the key length 30.
- Toemeh and Arumugam in [5]. This work described using of genetic algorithm (GA) to attack transposition cipher systems. The fitness values calculated based on bigrams, trigrams, and weights with some modification. Experimental results from this work showed that the GA successful to recover the secret key with length up to 15 when the ciphertext length is more than 1000 characters.
- Hameed and Hmood in [6]. In this paper, the cryptanalysis of transposition cipher based on PSO had introduced. This work based on diagram and trigram frequency analysis to calculate the fitness values. Experimental results illustrated that the PSO is an efficient way to attack the transposition cipher system. Tests from this work proved that the PSO was successful to determine the secret key with length up to 10 and can recover about 10 elements from the key length 15.
- Chen and Rosenthal in [7]. This work applied Markov Chain Monte Carlo (MCMC) to break transposition ciphers and substitution-plus-transposition ciphers. For the fitness values, bigram frequencies were used. This work was successful to determine the secret key up to 20 and about 85% of the key length 30.
- Heydari et al in [8]. This paper used of an improved GA to break transposition cipher system based on a novel fitness function. The proposed fitness based on the most common bigrams and trigrams. This work presented an improvised GA to decrypt transposition cipher with long key lengths. The Experimental results from this work indicated that uses of the improved fitness function were highly robust technique in cryptanalysis transposition cipher with long key up to 25 elements.
- Jassim in [9]. In this research, the cryptanalysis of simple transposition cipher system based on classical PSO and an improved PSO had presented. Proposed method in this work was applying the swap process after generate a new neighbour candidate key even when here is no improvement in results. The proposed method got 9% a percentage improvement in the number of detections keys as compared with the original method. The experimental results from this paper showed that the proposed method successful to recover about 88% from the key length 15 and recover 86% from key length 25.

1.2. Discussion of related work

There are many prior works focused on cryptanalysis transposition cipher systems based on different methods. The first and second works which presented by [3] and [4] attempted to attack transposition cipher with key up to 25, [3] was successful to recover key with length up to 25 but [4] was successful to recover only 20 elements. The prior works: [5] and [6] attacked transposition cipher with maximum key length 15 but for longer key, the performance weren't available. For the prior work that presented by [7], MCMC method successful to recovered about only 24 elements from key length 30 and the prior work of [8] was successful to break transposition cipher with long key up to 25 elements. Finally, the prior work [9] attacked transposition cipher with maximum key length 25 and successful to recover about 21 elements.

Table 1. Results of related works.

Keylen	Prior work	A.Txt	N.R.K	T
15	[3]	255	15	91
	[4]	1000	13	NR
	[5]	1000	15	100
	[7]	NR	NR	NR
	[8]	500	15	50
	[9]	1000	13	32
20	[3]	500	20	494
	[4]	1000	17	NR
	[5]	NR	NR	NR
	[7]	1000	19	0.88
	[8]	1000	20	150
	[9]	1000	16	58
25	[3]	750	25	829
	[4]	1000	21	NR
	[5]	NR	NR	NR
	[7]	NR	NR	NR
	[8]	2000	25	500
	[9]	1000	21	76
30	[3]	NR	NR	NR
	[4]	1000	25	NR
	[5]	NR	NR	NR
	[7]	2000	24	6.4
	[8]	NR	NR	NR
	[9]	NR	NR	NR

As shown in Table 1 that summarizes the results of related works where the adequate ciphertext length (A.Txt), number of recover key (N.R.K) and time consuming (T) in second to break transposition cipher system with different key lengths (Keylen). (NR) denotes to the not recovered field, we can see that the performance of the prior methods was limited and had lacked interest in time. Such as [8] was successful to apply his methods to break transposition cipher with long key up to 25 elements but for longer keys no performance was available and the time consuming was up to 500 seconds. Also the prior work of [4] was able to find only 25 elements from the key length 30 and the time consuming wasn't recovered. This research focuses on obtain best results in less time because there are many security systems based on transposition cipher try to change the key within specific periods. This work focuses on using of a new approach which is CLPSO to cryptanalysis transposition cipher. Also, this work implemented PSO algorithm in cryptanalysis transposition cipher and compared with the new approach to determine the difference between the original PSO and the new modified method. The modified algorithm can be applying to attack different types of cipher systems such as substitution and stream cipher systems.

2. Columnar transposition cipher

In transposition cipher systems, the cleartext splits into block which has a constant size and then the positions of characters in cleartext alters depending on the existing key to produce unreadable cleartext. Transposition cipher system provides diffusion by prevalence the information of plaintext based on the secret key to the obtainment of a wide range across the unreadable messages. The same characters in the cleartext have found in the ciphertext but in different sort [10]. Below simple example of columnar transposition cipher system:The cleartext is: "our strength lies in our unity" The secret key: (5 3 1 2 4)

5	**3**	**1**	**2**	**4**
o	u	r	s	t
r	e	n	g	t
h	l	i	e	s
i	n	o	u	r
u	n	i	t	y

Based on the length of key, the cleartext divides into blocks. In this example the key length is 5, so the cleartext is dividing into five blocks. When performing the encryption process, the block of letters in the 3rd column will be the first read, the block of letters in the 4th column will be the second read, then the block of letters in the 2nd column will be the third, the block of letters in the 5th column will be the fourth and the block of letters in the 1st column will be read latest. As a result of transposition cipher system processes, the ciphertext of the given cleartext is: "RNIOI SGEUT UELNN TTSRY ORHIU".

3. Particle swarm optimization (PSO)

PSO is a stochastic optimization, population-based evolutionary computer algorithm provided by James Kennedy and Russell C. Eberhart in 1995. Particle Swarm Optimization (PSO) incorporates swarming behaviors observed in bird flocking or swarms of bees or fish schooling. The PSO algorithm applies to think through various complex non-linear optimization problem because of the random nature of the PSO algorithm to obtain the optimal solution. The most important features of this algorithm are the simplicity of implementation, no complex mathematical functions, no costly mathematical, requires a few parameters and the ability to swiftly converge to the optimal solution [11]. PSO begins with randomly initialize population from a group of particles and each particle has its own position and velocity, these particles search through problem space. Each particle update's based on two values which are pbest and gbest, the first value (pbest) is the best values that obtains by the individual (particle) during the searches and the second value (g best) is the global best value of particle in the entire swarm. The following equations represent the evolutions of velocity (Vijt+1) and position (Xijt+1) for each particle [12]:

$$V_{ij}^{t+1} = V_{ij}^{t} + C1 * r1_{j}^{t} * \left[Pbest_{i}^{t} - X_{ij}^{t}\right] + C2 * r2_{j}^{t} * [Gbest - X_{ij}^{t}] \quad (1)$$

$$X_{ij}^{t+1} = X_{ij}^{t} + V_{ij}^{t+1} \quad (2)$$

Where; V: velocity of the individual (particle), i: represent the single individual(particle), j: the dimension in the problem space, t: represent the number of iteration, X: position of the individual (particle), r1, r2: represent spontaneous number between (-1 and 1) or (0 and 1), C1, C2: represent the acceleration parameter.

3.1. Parameters of PSO

The performance of PSO algorithm based on some parameters, some of these parameters have a significant and effective impact on the efficiency of the PSO algorithm and the others have small or no effect on the performance of PSO algorithm. The main PSO parameters are: size of the swarm which represents the number of particle in the swarm, velocity components which are lies in the range [-Vmax, Vmax] and use to limit the maximum jump for the single step of each particle. Cognitive and social

parameters which use to increase the speed of convergence and mitigation for local minima and the two random number(r1,r2) use to maintain the diversity of the population and finally the inertia weight (w) uses to take control the influence of the previous history of velocities on the current one.

4. Clustering

Clustering is a fundamental principle in the science of Data Mining. Clustering interests with grouping the similar objects together with each other and grouping the dissimilar objects together with each other, So the similar object and dissimilar object belong to different clusters. The main aims of clustering are determining feature groups and discover a new set of groups that are subject to evaluation [13]. The objects group into subsets in the same way that similar objects are grouping together and the dissimilar objects belong to another group. Regulate the objects into an efficient manner to realize the population that has sampled. There are many types of clustering methods. Generally, there are two main groups these are [14]:

- **Hierarchical methods**: create the clusters by recursively splitting the instances in either a bottom-up (Agglomerative hierarchical) or a top-down (Divisive hierarchical) fashion.
- **Partitioning methods**: convey objects by moving them from one cluster to another. The Partitioning methods require from the user to determine the number of clusters. In partitioning methods, to achieve global optimality must be counting process of all possible partitions.

5. Fitness function

Most papers use fitness functions which combine unigram, bigram, and trigram frequency statistics to cryptanalysis transposition cipher systems. In this work, fitness function calculates based on Diagram, Trigram and Quad gram frequency statistics to break transposition cipher. The primary goal of attacker is get the secret key and this can be achieved by repeatedly swapping the positions of the candidate key until obtains the correct key and recover the ciphertext [15]. Equations (3and4) are used as the fitness function to cryptanalysis transposition cipher system.

$$A = \sum_{i=aa}^{zz} \left| P_i^{dtq} - C_i^{dtq} \right| \tag{3}$$

Where P_i^{dtq} denotes the aggregate frequencies of plaintext (Diagram, Trigram, and Quadgram), C_i^{dtq} denotes the aggregate frequencies of ciphertext (Diagram, Trigram and Quadgram) and A denotes to the coincidence of frequencies between plaintext and ciphertext.

$$Fitness = P + (1 - A) \tag{4}$$

Equation (4) illustrated that the fitness value is best when the value of P is maximum and the value of A is minimum.

Algorithm (1): fitness function

- while I <= The maximum length of text do:
- Calculation the diagram, trigram and quad grams' frequencies of plaintext.
- Calculation the aggregate frequencies of plaintext (P).
- Calculation the diagram, trigram and quad grams' frequencies of ciphertext.
- Calculation the aggregate frequencies of ciphertext (C).
- Calculate the coincidence of frequencies between plaintext and ciphertext (A).
- Fitness value= P + (1-A).
- End while.

6. The proposed algorithm (CLPSO)

As we described in section 4, the main purpose of the clustering process is grouping the same elements in the same cluster and the other elements are grouping to the different cluster. Clustering based particle swarm optimization algorithm (CLPSO) which is suggested as the proposed algorithm in this research employs the partitioning based cluster methods in the population of the PSO algorithm. As we described in section 3, PSO algorithm begins with randomly initialized population from a group of particles, for the proposed system**,** use of large population size (particles) for initial population of the PSO algorithm in order to obtain large candidates solutions, since each particle in population represent candidate solution. After some iteration (threshold), typically about 10% from maximum iteration, apply clustering method by assign a number of clusters (k) to PSO population and spreading the particles in these clusters. Distribution particles on the clusters based on the fitness of each particle, Since the partitioning based cluster methods groups the similar elements in the same cluster and the different element to another cluster, So the particles with similar fitness function group together in the same cluster and the particles with different fitness function group to different clusters. As a result of this, the proposed algorithm creates a number of clusters, the first cluster contains the most preferable particles and the second cluster contains the second most preferable particles cluster and soon so that the last clusters contains the most worse particles. After grouping the particles into clusters, descend sorting of particles in each cluster based on the fitness values of each particle. Allow particles to move from one cluster to another by sharing information between particles in the same cluster and particles in a different cluster. After some generation reduce population size by discarding the worst particles from each cluster depending on the deletion percentage, for example, if we have 5 clusters, the deletion percentage from k1 is 50% and the deletion percentage from k2 is 80% and the deletion percentage from k3 is 80% and the deletion percentage from k4 is 95% and the deletion percentage from k5 is 95%. After reducing the population, the new population continues the search for a new solution until maximum iterations are hookup.

6.1. Initialization, Evaluation and termination condition

In CLPSO algorithm, the initialization starts randomly from a large number of particles (population size) to cover the entire search space then the population is clustering into several swarms based upon the proximity of the particles in the problem space. Assign position and velocity for each particle and each particle represent the candidate solution. For the optimal initial population, the domain of the search space is bounded to a minimum and maximum velocity value, in this research the Vmin = -4 and the Vmax= 4. Each particle search for an optimal solution by updating the position and velocity according to the following equations:

$$V_{ij}^{t+1} = V_{ij}^{t} + C1 * r1_{j}^{t} * \left[Pbest_{i}^{t} - X_{ij}^{t}\right] + C2 * r2_{j}^{t} * \left[Gbest - X_{ij}^{t}\right] + C3 * r3_{j}^{t} * \left[Gbest - X_{ij}^{t}\right] \quad (5)$$

$$X_{ij}^{t+1} = X_{ij}^{t} + V_{ij}^{t+1} \quad (6)$$

The initial population in the swarm has a significant impact on the control of exploration and exploitation in search space to find the best solution. To evaluate particle, the fitness value for each particle calculates at the beginning of generation and determined the local and global position (solution) in the swarm. For breaking transposition cipher, assign an integer number to the coordinates of Xijt which are referred to the permutation of the key and the population of particles constructs randomly. CLPSO algorithm terminates when the optimal solution has found or the maximum iteration has reached.

6.2. CLPSO algorithm parameters

CLPSO algorithm has some parameters which may effect on its performance. The basic parameters that have a significant impact on the efficiency of CLPSO algorithm are number of iterations, swarm size, inertia weight, velocity components and acceleration coefficients. In this research the number of iterations are [100-5000], swarm size is [100-5000], inertia weight is [0.4- 0.9] and acceleration coefficients are [0.5-2].

Algorithm (2): CLPSO Algorithm

- For each particle $i \in \{1,2, , np\}$// initial population.

- Initialize the position X_{ij}^t and velocity V_{ij}^t randomly.
- Survival of the fittest to evaluate the position of particles Fit(X_{ij}^t**)** // for each particle, the fitness value is calculating based on equation(4)**.**
- Save the best local position ($Pbest_i^t$) and best global position ($Gbest_i^t$).
- For each particle// evolve particles.
- Update the velocity and position based on equation (5 and 6).
- End for.
- While the termination criterion is not met do//main loop in CLPSO.
- While iteration = threshold do// Applying clustering method.
- For i=1 to numcluster do.
- Grouped the population size into some clusters.
- End for.
- For each particle// applying distribution process.
- Assign the ranking values for all particles based on the fitness.
- Distribute particles onto these clusters based their ranking.
- Update the velocity and position based on equation (5 and 6).
- Calculation fitness value.
- End for.
- End while.
- For each particle// reduce population.
- Ranking all particles based on the fitness value.
- Delete the worst particles.
- Redistribute the remain particle on the clusters.
- Update the velocity and position based on equation (1and 2).
- Calculation fitness value.
- End for.
- End while.

7. Use of CLPSO algorithm to cryptanalysis transposition cipher

PSO algorithm based clustering (CLPSO) is a new method in cryptanalysis. In this research, CLPSO uses to solve transposition cipher system. CLPSO algorithm starts with generate random numbers between {1, -1} for n particles, the length of this numbers must be equal to the length of the key. So, each particle represents the candidate key. For each particle, assign velocity value randomly, for this work the velocity values bounds between {4,-4}. The velocity value uses to enhance the exploration of the search space for each particle. Each particle uses to decrypt the transposition cipher and calculation the fitness value according to equation (4) to evaluate this particle and then each particle evolves based on three values (Pbest which represent the best solution for each particle(local) , Gbest which represent the best solution (particle) among all solution(particles) in the same cluster and GGbest which represent the best solution (particle) among all solution(particles) in all cluster which represent optimal solution). For Each particle, the velocity and position updates according to equation (5 and 6).

8. Experimental results

All experiences in this work performed on text using English alphabet and all experiences suppose that the entire text contains only capital letters. The space and all punctuation have removed. The proposed system implemented using Matlab program and performed using processor Intel (R) Core i3, RAM of 4.00 GB and 32-bit Windows 7 operating system. The CLPSO implemented successfully on different size of ciphertext provided to attack. In this research, the plaintext with different size are encrypted using transposition cipher with key length {10, 12…,35}and CLPSO algorithm is used to decrypt the ciphertext using ciphertext only attack and discover the encryption key in real time. Table 2 shows the outcomes of using CLPSO to break text encrypted using transposition cipher system with key length

(10),(12) and the text length are (250,500,1500,3000 and 5000) letters. The results of using CLPSO to break transposition cipher compared with PSO algorithm in the same circumstances. The results from the Table 2 illustrated that the performance of the both algorithms PSO and CLPSO are good to find the best fitness for different text size encrypted with key length 10 and 12. Also, the results from Table 1 showed that the CLPSO algorithm found the correct key for all tests in less than 1.7 second (real time), where the maximum consuming time for CLPSO algorithm to decrypt ciphertext with key length 10 is 1.6 second and for key length 12 is 1.66 when the text size (250). While the maximum consuming time for PSO is 44 second when the key length 10 and 115 second when the key length 12 when the text size (5000) and that time out of real time where the maximum time for real time is 40 second. Table 3 shows the outcome of using CLPSO and PSO algorithms to break text encrypted using transposition cipher system with key length (13 and 15) and the text length were (250,500,1500,3000 and 5000) letters.

Table 2. Results of using CLPSO and PSO to attack transposition cipher with key length (10 and 12).

Key size	Text length	CLPSO			PSO		
		Plaintext fitness	Best fitness	Time(sec)	Plaintext fitness	Best fitness	Time(sec)
10	250	2.46	2.46	1.6	2.46	2.22	36
	500	1.911	1.911	0.4	1.911	1.911	16.99
	1500	1.7	1.7	1.03	1.7	1.7	19.2
	3000	1.667	1.667	1.21	1.667	1.667	25.14
	5000	1.687	1.687	0.91	1.687	1.687	44.41
12	250	2.46	2.46	1.66	2.46	2.238	48.42
	500	1.911	1.911	1.47	1.911	1.632	94.71
	1500	1.7	1.7	1.44	1.7	1.438	105.2
	3000	1.667	1.667	1.3	1.667	1.43	113.6
	5000	1.687	1.687	1.18	1.687	1.435	115.5

Table 3. Results of using CLPSO and PSO to attack transposition cipher use the key length (13 and 15).

Key size	Text length	CLPSO			PSO		
		Plaintext fitness	Best fitness	Time(sec)	Plaintext fitness	Best fitness	Time(sec)
13	250	2.46	2.46	3.01	2.46	2.22	62.7
	500	1.911	1.911	2.3	1.911	1.6	53.9
	1500	1.7	1.7	2.08	1.7	1.38	133
	3000	1.667	1.667	2.7	1.667	1.3	106.3
	5000	1.687	1.687	3.3	1.687	1.32	193.4
15	250	2.46	2.46	15.68	2.46	2.122	96.1
	500	1.911	1.911	12.06	1.911	1.59	102.5
	1500	1.7	1.7	10.66	1.7	1.366	158.5
	3000	1.667	1.667	11.16	1.667	1.31	126.3
	5000	1.687	1.687	11.88	1.687	1.27	165.2

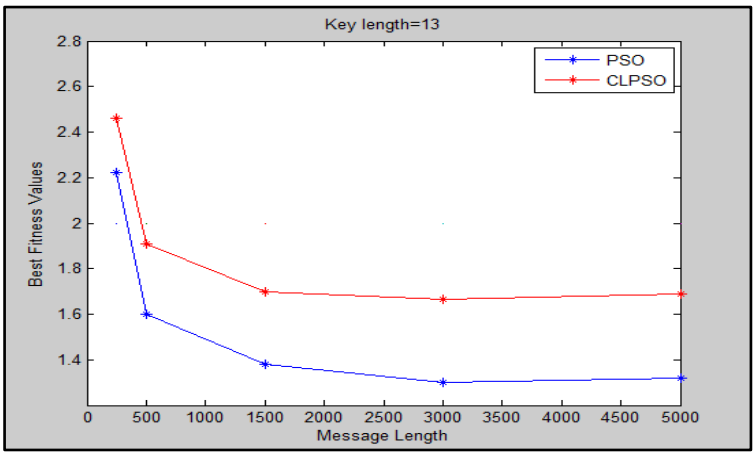

Figure 1. Plot curve of the fitness values of CLPSO and PSO with key length (13).

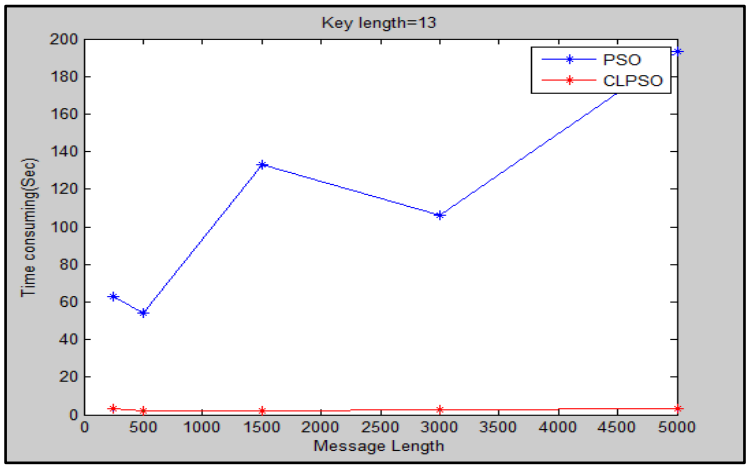

Figure 2. Plot curve of the time consuming of CLPSO and PSO with key length (13).

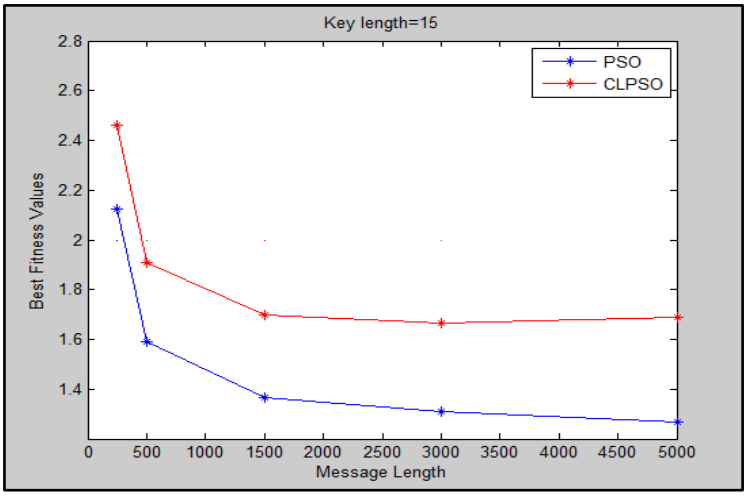

Figure 3. Plot curve of the fitness values of the CLPSO and PSO with key length (15).

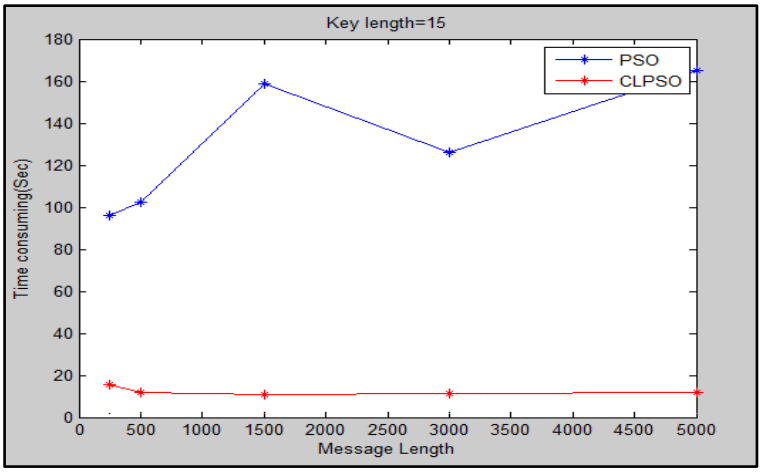

Figure 4. Plot curve of the time consuming by CLPSO and PSO with key length (15).

From Table 3 and the plot curves in Figure 1, Figure 2, Figure 3 and Figure 4, the results illustrated that the CLPSO algorithm found the correct key for all ciphertext length when the key length (13 and 15) and the time taken to the best results is very good where the maximum time is less than 16 second (real time). While, the performance of the PSO algorithm is different between good and bad for example, the performance of the PSO algorithm to attack text with length (500) encrypted with key length (13) is good but the performance of the PSO algorithm worse when attack the others and the maximum time was less than 193 second which represent much when compared with maximum time for CLPSO which was 16 second. Table 4 and Table 5 showed the outcome of using CLPSO and PSO algorithms to break text encrypted using transposition cipher system with key length (20,25,30 and 35) respectively and the text length were (250,500,1500,3000,5000) letters.

Table 4. Results of using CLPSO and PSO to attack transposition cipher use the key length (20, 25).

Key size	Text length	CLPSO			PSO		
		Plaintext fitness	Best fitness	Time(sec)	Plaintext fitness	Best fitness	Time(sec)
20	250	2.46	2.46	20.16	2.46	1.96	112.01
	500	1.911	1.911	18.01	1.911	1.54	125.2
	1500	1.7	1.7	16.06	1.7	1.33	120.02
	3000	1.667	1.667	16.00	1.667	1.32	128.18
	5000	1.687	1.687	16.8	1.687	1.266	170.91
25	250	2.46	2.46	25.06	2.46	1.92	222.32
	500	1.911	1.911	23.88	1.911	1.5	168.00
	1500	1.7	1.7	23.02	1.7	1.32	210.54
	3000	1.667	1.667	21.8	1.667	1.3	235.55
	5000	1.687	1.687	22.9	1.687	1.25	268.11

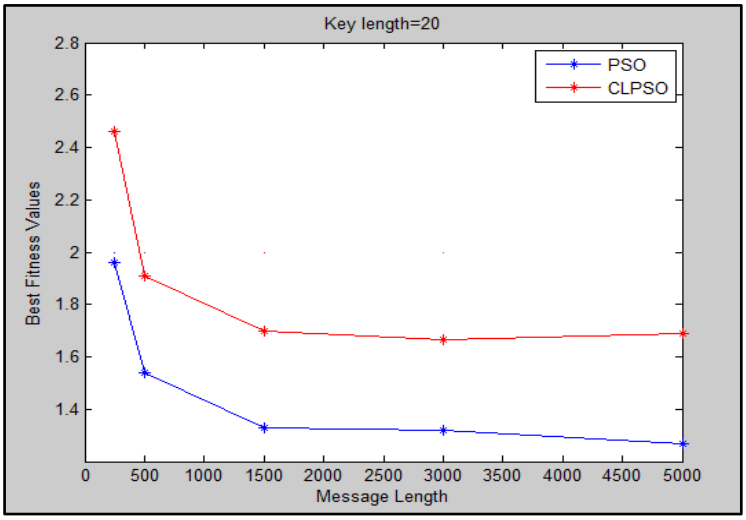

Figure 5. Plot curve of the fitness values of the CLPSO and PSO with key length (20).

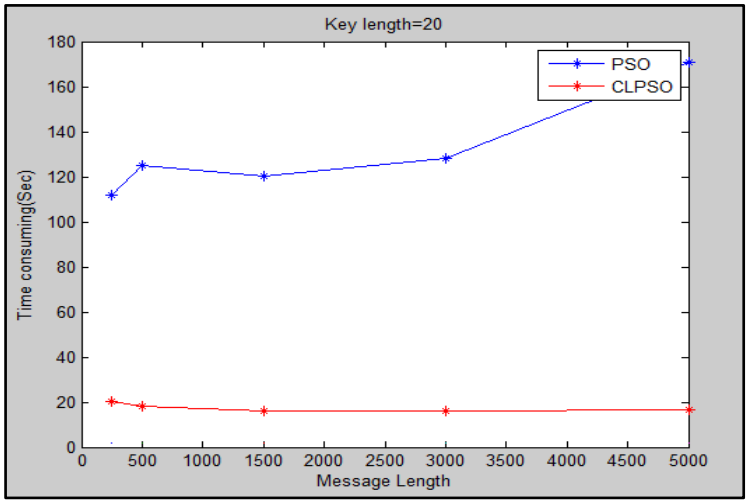

Figure 6. Plot curve of the time consuming by CLPSO and PSO with key length (20).

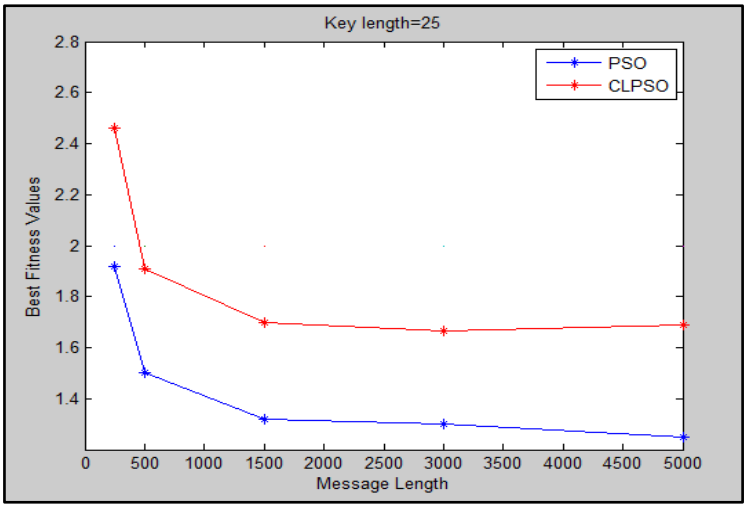

Figure 7. Plot curve of the fitness values of the CLPSO and PSO with key length (25).

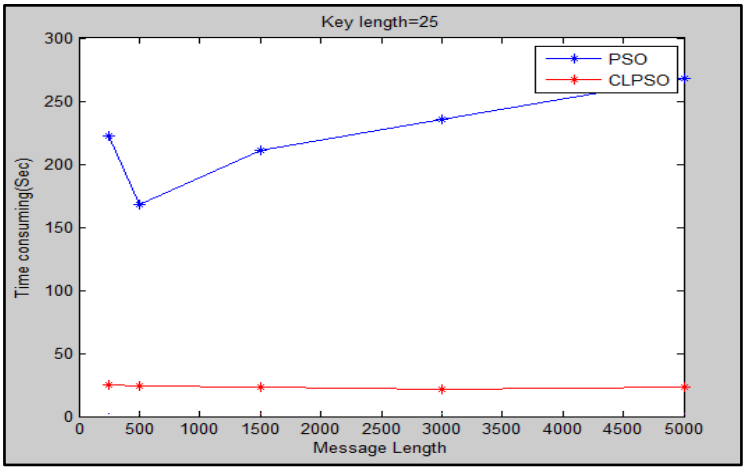

Figure 8. Plot curve of the time consuming by CLPSO and PSO with key length (25).

Table 5. Results of using CLPSO and PSO to attack transposition cipher use the key length (30, 35).

Key size	Text length	CLPSO			PSO		
		Plaintext fitness	Best fitness	Time(sec)	Plaintext fitness	Best fitness	Time(sec)
30	250	2.46	2.46	40.54	2.46	1.89	256.8
	500	1.911	1.911	38.67	1.911	1.48	248.37
	1500	1.7	1.7	33.01	1.7	1.27	233.58
	3000	1.667	1.667	32.98	1.667	1.3	246.91
	5000	1.687	1.687	32.33	1.687	1.26	289.1
35	250	2.46	2.22	58.23	2.46	1.81	432.5
	500	1.911	1.81	53.07	1.911	1.41	350.00
	1500	1.7	1.7	38.55	1.7	1.25	355.84
	3000	1.667	1.667	39.68	1.667	1.27	380.54
	5000	1.687	1.687	42.03	1.687	1.23	420.1

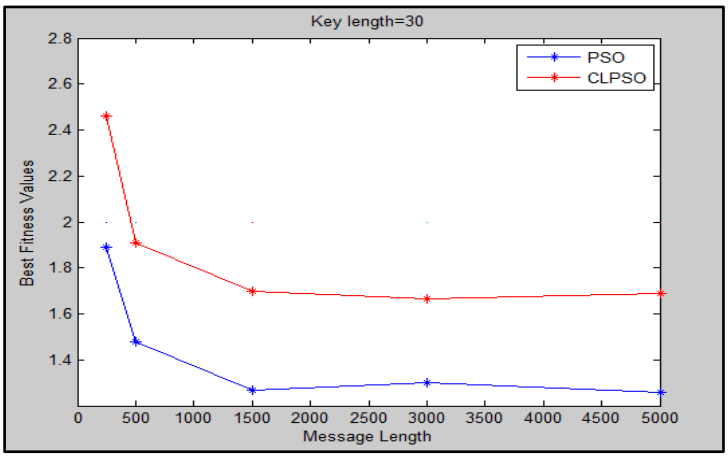

Figure 9. Plot curve of the fitness values of the CLPSO and PSO with key length (30).

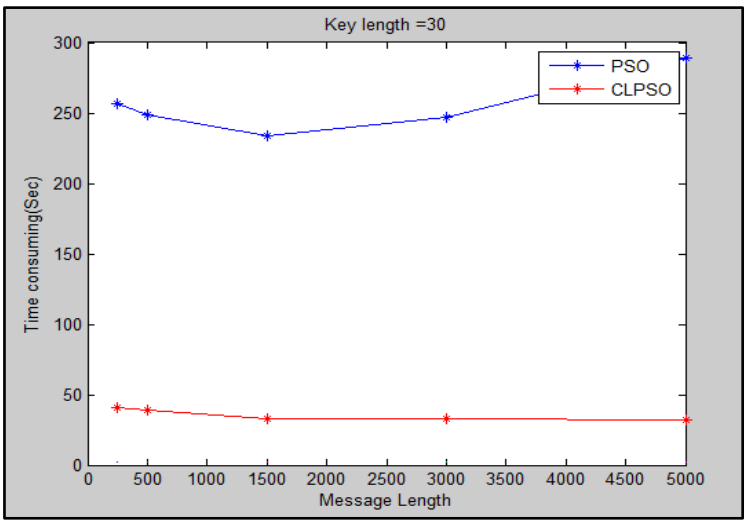

Figure 10. Plot curve of the time consuming by CLPSO and PSO with key length (30).

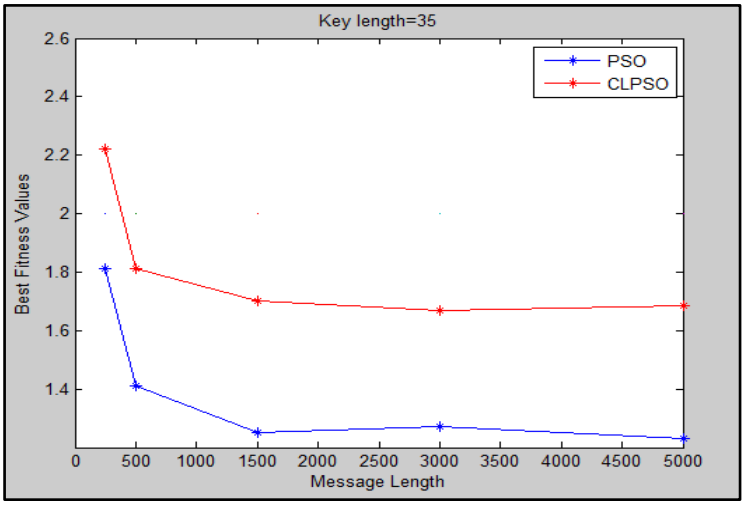

Figure 11. Plot curve of the fitness values of the CLPSO and PSO with key length (35).

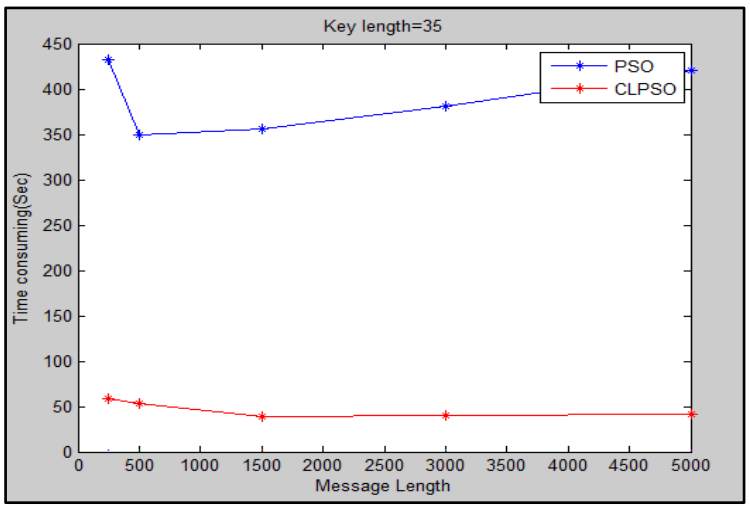

Figure 12. Plot curve of the time consuming by CLPSO and PSO with key length (35).

The results from Table 4, Table 5 and figures from (Figure 5 - Figure 12) show that the performance of the CLPSO algorithm was much better than PSO because the best fitness that obtained by applying CLPSO to attack different sizes of text encrypted using long key size (20,25,30 and 35) was very close to the fitness of the original text, where the number of recover key element was 100% except the state when the key size was 35 and text size 250 and 500 where the recover key element was 90% and 93% respectively. The consuming time for CLPSO was very reasonable, where the maximum consuming time was 20.1second when the key length 20, 25.1 second when the key length 25, 40.5 when the key length 30 and 58.3 second when the key length 35 while the maximum consuming time by PSO algorithm up to 420 second with bad accuracy.

9. Conclusions

This research focused on using a new approach which was the use of particle swarm optimization with clustering algorithm (CLPSO) to solve transposition cipher system with great interest in time and the results were compared with the original PSO algorithm. The PSO and CLPSO algorithm were used to recover the secrete key in order to obtain the plaintext. The calculation of the fitness function was based on findings from the diagrams, trigrams and quadgrams frequencies for the candidate plaintext. The converges of the CLPSO algorithm to solutions was very quick and the results taken from Table 1, Table 2, Table 3 and Table 4 show that the CLPSO was very efficient to attack long key transposition cipher.

The experimental results showed that the use of clustering algorithm with PSO algorithm made the performance of the PSO algorithm much better in cryptanalysis. The current adopted new method had less number of trying to visit keys that required to break the message than in the original PSO such as for the case when the key length=30, CLPSO found all elements of the secret key for different cipher length with the time no more than 41 second. The number of iteration that required finding the best solution using CLPSO algorithm was less than the number of iteration that required to find the best solution using PSO algorithm.

References

[1] Stamp M and Richard M L 2007 *Applied Cryptanalysis: Breaking Ciphers in The Real World* (New Jersey: John Wiley & Sons) pp 1-7
[2] Saveetha P, Arumugam S and Kiruthikadevi K 2014 Cryptography and the optimization heuristics techniques *Int. J. Adv. Res. Comp. Sci. Soft. Eng.* **4**
[3] Giddy J P and Safavi R N 1994 Automated cryptanalysis of transposition ciphers *Comp. J.* **37**
[4] Dimovski A and Gligoroski D 2003 Attacks on the transposition ciphers using optimization heuristics *Proc. on ICEST 2003 (Sofia)*
[5] Toemeh R and Arumugam S 2007 Breaking transposition cipher with genetic algorithm *Electronics and Electrical Engineering* **79**
[6] Hameed S M and Hmood D N 2010 Particles swarm optimization for the cryptanalysis of transposition cipher *J. Al-Nah. Univ.* **13**
[7] Chen J and Rosenthal J S 2012. Decrypting classical cipher text using Markov chain Monte Carlo *Stat. Comp.***22**
[8] Heydari M, Shabgahi G L and Heydari M M 2013 Cryptanalysis of Transposition Ciphers with Long Key Lengths Using an Improved Genetic Algorithm *Wor. Appl. Sci. J.* **21**
[9] Jassim M K 2017 Improved PSO algorithm to attack transposition cipher *Eng. Techol. J.* **35** 144-149
[10] Stallings W 2010 *Cryptography and Network Security: Principles and Practice*. 5th ed Upper Saddle River (NJ: Prentice Hall) p 54
[11] Xia X, Gui L, He G, Xie C, Wei B, Xing Y, Wu R and Tang Y 2018. A hybrid optimizer based on firefly algorithm and particle swarm optimization algorithm *J. Comp. Sci.* **26**
[12] Panigrahi B K, Das S, Suganthan P N and Dash S S 2010 Swarm, Evolutionary and Memetic Computing. *1st Int. Conf. on SEMCCO 2010* vol 6466 (Berlin)
[13] Ng R T and Han J 2002 CLARANS: A method for clustering objects for spatial data mining *IEEE Tran. Know. Data Eng.* **14**
[14] Chitra K and Maheswari D 2017 A comparative study of various clustering algorithms in data mining Int. J. Comp. Sci. Mob. Comp. **6**
[15] Ali F H 2015 *Improving exact and local search algorithms for solving some combinatorial optimization problems* [dissertation] Iraq: University of Al-Mustansiriyah