Cipher System using Artificial Bees Colony Algorithm

Assistant Prof. Dr. Ziyad Tariq Mustafa [1] and Rajaa Ahmed[2]

[1] Head department Computer Science-College of Science-University of Diyala

[2] College of Science-University of Diyala

Abstract

Classical ciphers were first used hundreds of years ago. As far as security is concerned, they are no match for today's ciphers; however, this does not mean that they are any less important to the field of cryptology. A cipher system is one for which applying encryption algorithm to plaintext produces cipher text. The weakness with this strategy is that cipher text frequency distribution is not significantly altered by the encryption process. Swarm Intelligence (SI) is an artificial intelligence technique based on the study of collective behavior in decentralized and self- organized systems. This paper uses one of swarm intelligence optimization algorithms called Artificial Bees Colony (ABC) algorithm as an intelligent cryptographic tool to enhance the security of simple cipher system. An image data are used as the seed keys for the proposed cipher system, while the ABC algorithm is used to obtain the randomness for these keys. The results are successfully tested with randomness tests.

Keywords: Cipher System, Swarm intelligence, Bees algorithm, ABC Algorithm.

نظام تشفير بسيط باستخدام خوارزمية ABC

1- **أ.م.د زياد طارق الطائي** / رئيس قسم علوم الحاسبات/ كلية العلوم / جامعة ديالى

2- **رجاء احمد علي** / قسم علوم الحاسبات

DIYALA JOURNAL FOR PURE SCIENCES

Cipher System using Artificial Bees Colony Algorithm

Assistant Prof. Dr. Ziyad Tariq Mustafa [1] and Rajaa Ahmed[2]

الخلاصة

تم أستخدام التشفير التقليدي منذ مئات السنين ، ونظراً لأهمية الأمنية فأن هذا التشفير لا يتوافق مع التشفير الحالي. ولكن هذا لا يعني أن التشفير التقليدي أقل أهمية بالنسبة لعلم التشفير. نظام التشفير هو الذي يطبق خوارزمية تشفير على نص واضح ليولد نص مشفر. أن نقطة ضعف هذه الاستتراجية هو أن التوزيع الترددي لحروف النص المشفر لا يتغير كثيراً بعملية التشفير. أن ذكاء السرب هو تقنية ذكاء أصطناعي تستند الى السلوك الجماعي في أنظمة الترتيب الذاتي غير المركزية. يستخدم هذا البحث واحدة من خوارزميات مفاضلة ذكاء السرب تسمى خوارزمية مستعمرة النحل الأصطناعية كأداة تشفير ذكية لغرض تحسين أمنية نظام التشفير البسيط. تم أستخدام بيانات الصورة كمفاتيح اولية لنظام التشفير المقترح بينما تم أستخدام خوارزمية مستعمرة النحل الأصطناعية للحصول على عشوائية هذه المفاتيح . علماً ان النتائج تم أختبارها بنجاح بواسطة اختبارات العشوائية

الكلمات المفتاحية: نظام التشفير, ذكاء السرب , خوارزمية النحل , خوارزمية مستعمرة النحل الاصطناعية.

Introduction

Cryptography is the science of using mathematics to encrypt and decrypt data. Cryptography enables us to store sensitive information or transmit it across insecure networks (like the Internet) so that it cannot be read by anyone except the intended recipient. While cryptography is the science of securing data, *cryptanalysis* is the science of analyzing and breaking secure communication. Classical cryptanalysis involves an interesting combination of analytical reasoning, application of mathematical tools, pattern finding, patience, determination, and luck [1].

Cryptography can be *strong* or *weak*. Cryptographic strength is measured in the time and resources it would require to recover the plaintext. The result of *strong cryptography* is cipher text that is very difficult to decipher without possession of the appropriate decoding tool [1].

Cryptography is the study of "mathematical" systems for solving two kinds of security problems: privacy and authentication. A privacy system prevents the extraction of information by unauthorized parties from messages transmitted over a public channel, thus assuring the sender of a message that it is being read only by the intended recipient. An authentication system prevents the unauthorized injection of messages into a public channel, assuring the receiver of a message of the legitimacy of its sender. A channel is considered public if its security is inadequate for the needs of its users. A channel such as a telephone line may therefore be considered private by some users and public by others. Any channel may be threatened with eavesdropping or injection or both, depending on its use [2]. A key is a value that works with a cryptographic algorithm to produce a specific cipher text. Keys are basically big numbers. Key size is measured in bits; the number representing a 1024-bit key is darn huge. In public key cryptography, the bigger the key, the more secure the cipher text. However, public key size and conventional cryptography's secret key size are totally unrelated. A conventional 80-bit key has the equivalent strength of a 1024-bit public key. A conventional 128-bit key is equivalent to a 3000-bit public key. Again, the bigger the key, the more secure, but the algorithms used for each type of cryptography are very different and thus comparison is like that of apples to oranges. While the public and private keys are mathematically related, it's very difficult [1]. Swarm Intelligence (SI) is the collective behavior of decentralized, self-organized systems, natural or artificial. The expression was introduced by Gerardo Beni and Jing Wang in 1989 [3], in the context of cellular robotic systems.SI systems are typically made up of a population of simple agents interacting locally with one another and with their environment. The inspiration often comes from nature, especially biological systems. The agents follow very simple rules, and although there is no centralized control structure dictating how individual agents should behave, local, and to a certain degree random, interactions between such agents lead to the emergence of "intelligent" global behavior, unknown to the individual agents. Natural examples of SI include ant colonies, bird flocking, animal herding, bacterial growth, and fish schooling ,bee colony[4].

In this paper one of swarm intelligence optimization algorithms called Artificial Bees Colony (ABC) algorithm is used as an intelligent cryptographic tool to enhance the security of simple transposition cipher.

Artificial Bee Colony (ABC) Algorithm

Artificial Bee Colony (ABC) is one of the most newly defined algorithms by Dervis Karaboga in 2005 [5], provoked by the intelligent behavior of honey bees. It is as easy as Particle Swarm Optimization (PSO) and Differential Evolution (DE) algorithms, and uses only common control parameters such as colony size and maximum cycle number. ABC as an optimization tool provides a population-based search method in which individuals called foods positions are customized by the artificial bees with time and the bee's aim is to discover the places of food sources with high nectar amount and at last the one with the highest nectar [6]. The colony of artificial bees contains three groups of bees: employed bees, onlookers and scouts. The employed bees bring loads of nectar from the food resource to the hive and may share the information about food source in the dancing area. These bees carry information about food sources and share them with a certain probability by dancing in a dancing area in the hive. The onlooker bees wait in the dances area for making a decision on the selection of a food source depending on the probability delivered by employed bees. The computation of probability is based on the amounts of the food source. The other kind of bee is scout bee that carries out random searches for new food sources. The employed bee of an abandoned food source becomes a scout and as soon as it finds a new food source it becomes employed again. Figure (1) shows a flow chart of the ABC algorithm [7].

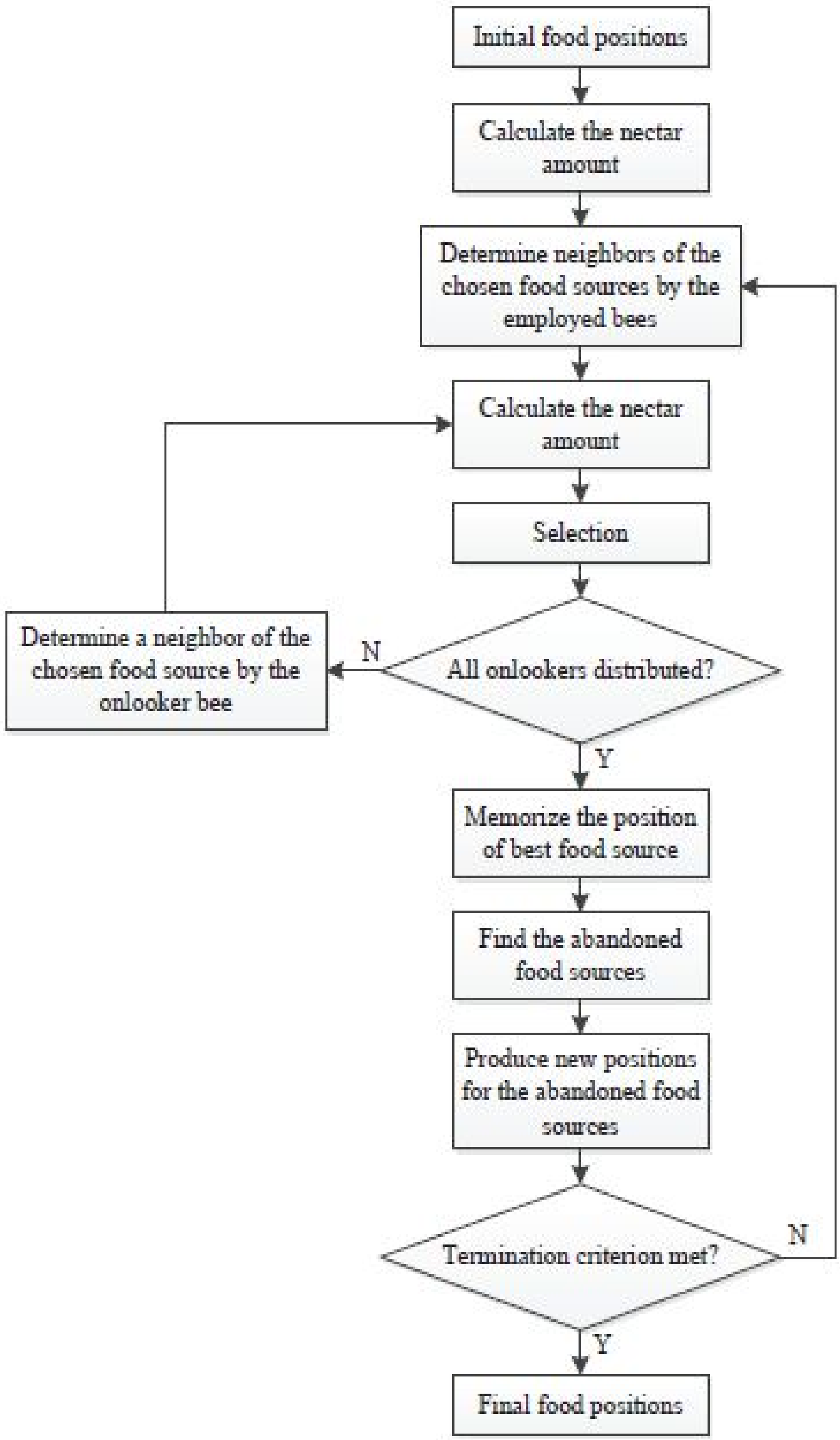

Fig.(1) Flow Chart of ABC Algorithm

For every food source, there is only one employed bee. Every bee colony has scouts that are the colony's explorers. The scouts are characterized by low search costs and a low average in food source quality. Occasionally, the scouts can accidentally discover rich, entirely unknown food sources. In ABC algorithm, the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. The number of the employed bees or the onlooker bees is equal to the number of solutions in the population. In the initialization phase, the ABC algorithm generates randomly distributed initial food source positions of SN solutions, where SN denotes the size of employed bees or onlooker bees. Each solution $\mathbf{x}_i (i=1,2,\dots,SN)$ is a n-dimensional vector. Here, n is the number of optimization parameters. Then each nectar amount fit_i is evaluated. In the employed bees' phase, each employed bee finds a new food source v_i in the neighborhood of its current source $\mathbf{x}_i$. The new food source is calculated using equation number (1).

$$v_{ij}=x_{ij}+\phi_{ij}(x_{ij}-x_{kj}) \qquad (1)$$

Where $k\in (1,2,\dots,SN)$ and $j\in(1,2,\dots,n)$ are randomly chosen indexes and $k\neq i$. ϕ_{ij} is a random number between $[-1,1]$. It controls the production of a neighbor food source position around x_{ij}. Then employed bee compares the new one against the current solution and memorizes the better one by means of a greedy selection mechanism. In the onlooker bees' phase, each onlooker chooses a food source with a probability which is related to the nectar amount (fitness) of a food source shared by employed bees. Probability is calculated using equation number (2).

$$Pi = fitsub/\sum_{i=1}^{SN} fitsub \qquad (2)$$

In the scout bee phase, if a food source cannot be improved through a predetermined cycles, called "limit", it is removed from the population, and the employed bee of that food source becomes scout. The scout bee finds a new random food source position using equation number (3).

$$x_i^j = x_{min}^j + rand()(x_{min}^j - x_{max}^j) \quad (3)$$

Where x_{min}^j and x_{max}^j are lower and upper bounds of parameter j, respectively, and rand is random numbers between (0,1). These steps are repeated through a predetermined number of cycles, called maximum cycle number (MCN), or until a termination criterion is satisfied [7].

The Proposed System

The block diagram of the proposed system is shown in figure (2).

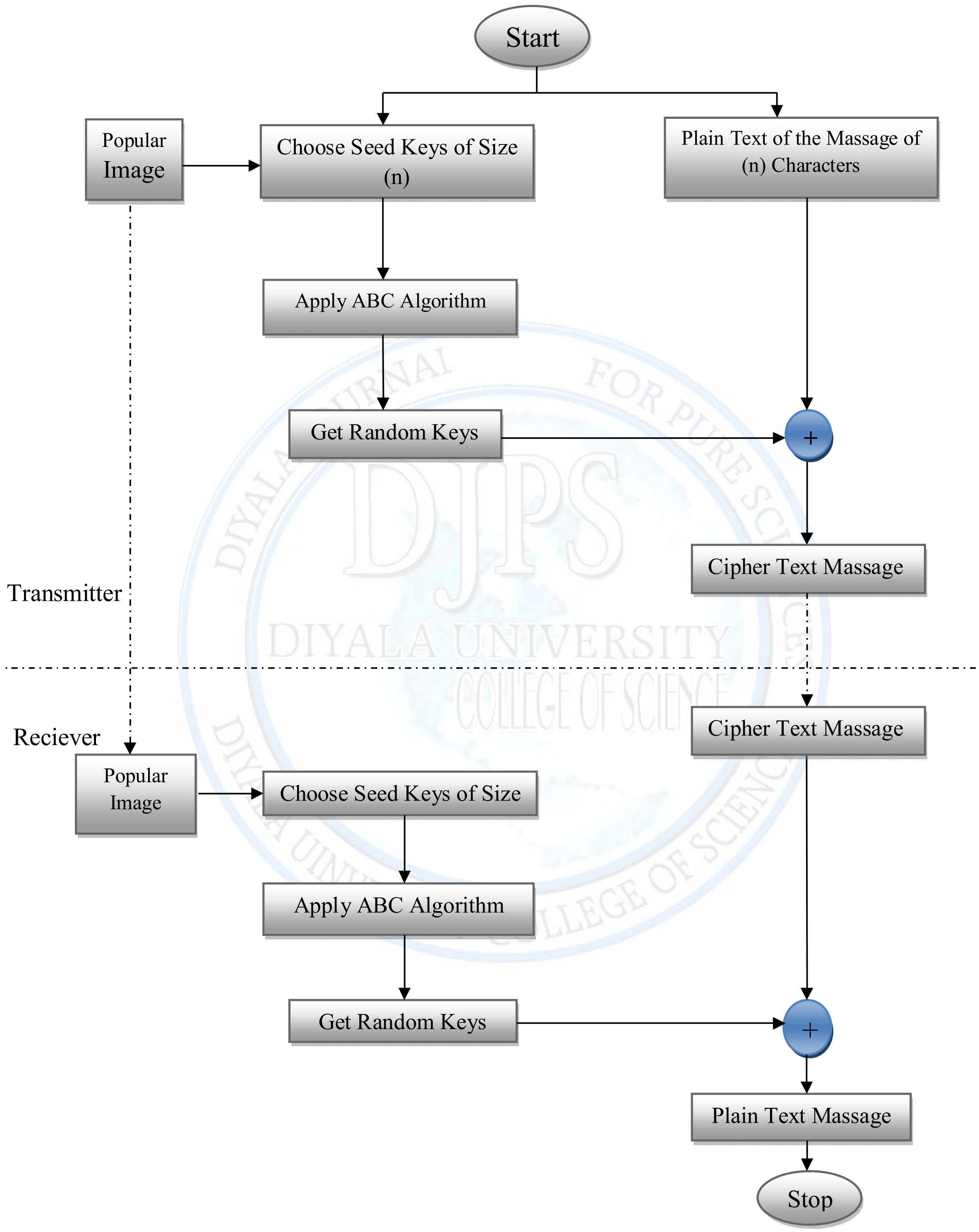

Figure (2) Block Diagram of the Proposed System

The proposed system starts with seeding initial keys from popular image data with size equals to number of characters in plain text massage. These keys are initial keys for ABC algorithm. ABC algorithm is used to give randomness for initial key as shown in algorithm (1).

Algorithm (1) ABC Algorithm
Input: Image data. Output: Cipher keys. 1: Initialize the population of solutions $x_{i,j}$ 2: Evaluate the fitness (Randomness of keys using tests) 3: Iteration=1 4: Repeat 5: Produce new solutions (food source positions) $\upsilon_{i,j}$ in the neighborhood of $x_{i,j}$ for the employed bees using the equation number(1). 6: Apply the greedy selection process between xi and υ_i 7: Calculate the probability values P_i for the solutions x_i by using the equation number (2). 8: Produce the new solutions (new positions) υ_i for the onlookers from the solutions x_i, selected depending on the probability. 9: Apply the greedy selection process for onlookers between x_i and v_i. 10: Determine the abandoned solution (source), if exists, and replace it with a new randomly produced solution x_i for the scout using the equation number (3). 11: Memorize the best food source position (solution) achieved so far 12: Iteration = Iteration +1 13: until Iteration = Maximum Cycle Number (MCN)

The parameters that are used in ABC algorithm are shown in table (1), and randomness fitness is shown in section (5).

Table (1): Parameters of ABC algorithm

Parameter	Symbol	Value
Number of bees	N	9
Maximum Cycle Number	MCN	20
Random numbers	φ	[-1,1]
Random numbers	rand	(0-1)
Limit	Limit	(0-255)

Results and Calculations

As an example, for ciphering the statement (**ONE THING THAT I WOULD LIK TO PRESENT HERE ARE SOME SIMPLE ANALYSIS TECHNIQUES THAT CAN BE USE TO HELP**), table (2) shows the values of random keys which are obtained by ABC algorithm for (20) iterations.

Table (2) Values of Random Keys which are created by ABC Algorithm

Initial Key from Image Data	**121**	**54**	**50**	**77**	**210**	**104**	**25**	**125**	**0**	**23**	**90**	**24**	**120**	**25**	**65**	**26**	**100**
Iteration 1	230	230	216	180	130	77	50	210	26	27	30	28	62	29	100	30	90
Iteration 2	80	120	200	190	10	62	63	160	50	31	120	32	60	33	180	34	230
Iteration 3	255	200	104	120	63	120	213	104	101	35	80	36	30	37	75	38	100
Iteration 4	80	210	190	125	108	130	120	200	26	39	40	40	25	41	0	42	8
Iteration 5	100	200	63	160	210	30	255	120	90	43	90	44	100	45	8	46	25

Iteration 6	95	140	80	100	190	63	230	90	40	47 180	48 82	49 100	50 60
Iteration 7	125	150	200	195	90	140	150	95	62	51 90	52 35	53 95	54 100
Iteration 8	200	70	108	80	110	125	85	8	80	55 77	56 95	57 100	58 120
Iteration 9	120	100	95	104	65	210	125	25	75	59 230	60 210	61 95	62 80
Iteration 10	95	120	110	200	100	95	85	63	90	63 73	64 210	65 200	66 90
Iteration 11	180	95	210	104	95	110	73	75	95	67 80	68 230	69 210	70 101
Iteration 12	255	180	195	62	100	210	95	80	110	71 95	72 200	73 255	74 100
Iteration 13	180	220	200	70	95	106	104	25	60	75 80	76 190	77 95	78 120
Iteration 14	190	110	180	90	213	95	106	60	80	79 90	80 120	81 210	82 213
Iteration 15	225	213	190	110	150	95	110	25	75	83 85	84 95	85 100	86 255

Continue of Table (2)

Iteration 16	195	210	180	200	75	80	95	60	85	87 100	88 110	89 185	90 213
Iteration 17	200	185	98	80	100	90	210	75	100	91 100	92 200	93 95	94 180
Iteration 18	190	213	110	85	95	100	225	80	185	95 95	96 185	97 110	98 213
Iteration 19	210	200	190	87	75	110	200	90	190	99 200	100 190	101 90	102 130
Iteration 20	195	110	200	95	180	225	213	100	210	103 255	104 180	105 210	106 213

Random Keys after iteration number twenty are exclusive ored with plain text characters using equation (4).

$$C = P \ominus K \; mod \; 26 \qquad \text{….. (4)}$$

Then the cipher text of (**ONE THING THAT I WOULD LIK TO PRESENT HERE ARE SOME SIMPLE ANALYSIS TECHNIQUES THAT CAN BE USE TO HELP**) is: (***KJM VXAPQ OZAJ K SGFPF CSM OQ HJIUKIA XAIW RFK QFOE NGZZHG YHSDJKS BKATSZSCG KMIZ PCL TH CXK VG UKGG***).

Figure (3) shows the relation between the values of random keys and number of iterations, for one character.

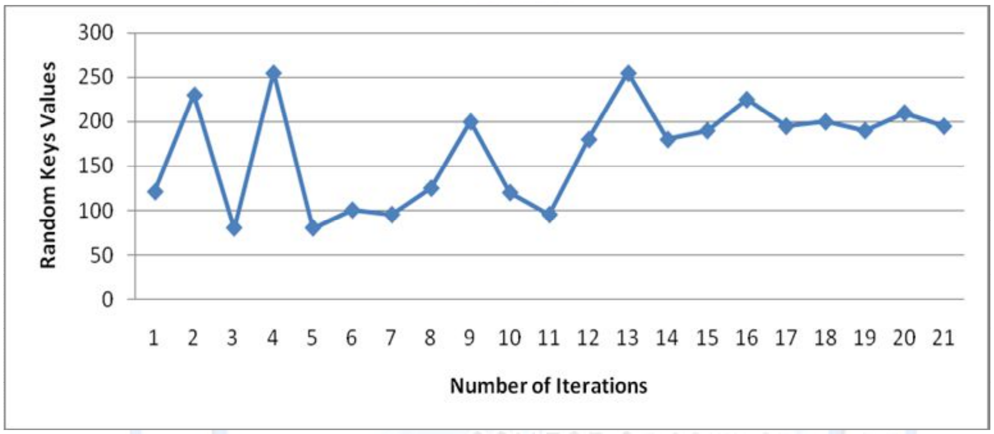

Figure (3) Relation between the Values of Random Keys and Number of Iterations, for One Character.

Table (3) shows the time required for each iteration.

Table (3) Required Time for ABC Algorithm

Iteration	Time Required (sec)
1	0.30
2	0.25
3	0.34
4	0.25
5	0.20
6	0.30

7	0.30
8	0.30
9	0.35
10	0.40
11	0.40
12	0.42
13	0.40
14	0.50
15	0.50
16	0.50
17	0.45
18	0.50
19	0.45
20	0.50

Tests

Table (4, 5, and 6) shows the results of randomness tests with significance of 5% in binary format on random key values that are generated by ABC algorithm.

Table (4) Randomness Tests on Random Keys Values with (5) Iteration

Randomness Tests on the Binary Sequences			
Name of the test	**Result**	**Real Value**	**Standard Value**
Frequency Test	pass	2.112	<=3.84
Serial Test	Pass	5.632	<=7.81
Poker Test	Pass	10.729	<=11.1

Run Test	Fail	8.523	<=7.724
Autocorrelation Test	Fail	Fail	

Table (5) Randomness Tests on Random Keys Values with (10) Iteration

Randomness Tests on the Binary Sequences			
Name of the test	**Result**	**Real Value**	**Standard Value**
Frequency Test	pass	1.482	<=3.84
Serial Test	Pass	3.679	<=7.81
Poker Test	Pass	8.341	<=11.1
Run Test	Pass	4.724	<=7.724
Autocorrelation Test	Fail	Fail	

Table (6) Randomness Tests on Random Keys Values with (15) Iteration

Randomness Tests on the Binary Sequences			
Name of the test	**Result**	**Real Value**	**Standard Value**
Frequency Test	pass	1.232	<=3.84
Serial Test	Pass	2.974	<=7.81
Poker Test	Pass	6.811	<=11.1
Run Test	Pass	3.295	<=7.724
Autocorrelation Test	Pass	Pass	

Conclusions

From this work, several conclusions can be drawn as follows:

1- The results presented in this work show that the ABC algorithm is a powerful algorithm that succeeds in generating random keys with suitable time for each iteration as shown in table(3).

2- Total time required by ABC algorithm can be reduced by minimizing number of iterations and randomness tests proves that (10) iterations are enough as shown in tables (4, 5, and6).

3- Depending on [9], PSO algorithm needs to be run (30) times in order get enough randomness, while ABC algorithm needs to be iterated (10) iterations for the same parameters (string of (101) characters, time required to execute, and laptop with processor core I3).

References

1. Internet survey, **"The Basics of Cryptography"** , http://www.diserio.com/ cryptography.html , accessed at 2/6/2012.
2. Whitfield Diffie and Martin E. Hellman, **"New Directions in Cryptography"**, IEEE Transaction on Information theory workshop, 1975.
3. Nivember 1976. Beni, G., Wang, J. "**Swarm Intelligence in Cellular Robotic Systems"**, Proceed. NATO Advanced Workshop on Robots and Biological Systems, Tuscany, Italy, 1989.
4. Internet survey, **"Swarm_intelligence "**, http://en.wikipedia.org/wiki, accessed at 18/6/2012.
5. Dervis Karaboga , Bahriye Basturk, **“A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm”** International Journal of Global Optimization , pp 459-471 , volume -39, Issue- 3, November 2007.

6. Manish Gupta, Govind Sharma, **"An Efficient Modified Artificial Bee Colony Algorithm for Job Scheduling Problem"** International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-1, Issue- 6, January 2012
7. Internet survey, **"solving-multi objective-optimization-problems using artificial bee"**, http://www.deepdyve.com/lp/hindawipublishing- corporation/r25Nf3cCA4, accessed at 22/6/2012.
8. Ahmed Hussein Ali, **"A Proposed Cryptographic Key Generator Algorithm"**, Ph.D. Thesis in Computer Science. Iraqi Commission for Computers and Informatics, Informatics Institute for Postgraduate Studies, Iraq,2013.
9. Mark Rodgers, **"Random Numbers and Their Effect on Particle Swarm Optimization"**,On_Line paper, http://ncre.ucd.ie/COMP30290/Crc2006/rodgers.pdf , Accessed at 15/3/2013.