Efficient Speech Scrambling Using Ant Mating Optimization

[1]N.Malathy[*] , [2]R.Tamilroja , [3]A.S.Nisha and , [4]T.Anuja

[1]Asisstant Professor (senior Grade), [1]Department of Information Technology, Mepco Schlenk Engineering College, Sivakasi.

[2]Asisstant Professor, [2]Department of Computer Science and Engineering, Jeppiaar SRR Engineering College, Chennai.

[3]Asisstant Professor, [3]Department of Computer Science and Engineering, Arunachala College of Engineering for Women, Nagercoil.

[4]Asisstant Professor, [4]Department of Information Technology, Jeppiaar Engineering College, Chennai.

Abstract

Ensuring security in any communication network through which vital data is exchanged is a critical concern. The usage of voice scrambling technologies is among the reasonable alternatives for maintaining safety for speech. When utilized to strengthen the hybrid of more than one approach, the chaotic system used in security has qualities that make it a better option for scrambling voice signals, and the optimization technique can deliver flawless performance. In this article, we introduce a methodology that employs Ant Mating optimization (AMO) as an optimization approach. The assessment metrics show that the optimization method's result surpasses the other methods in the comparison, such as chaotic maps and hybrid chaotic maps.

Keywords AMO, Security, Speech Scrambling.

Introduction

Every communication system's security has grown increasingly popular in recent years. As a result, we must safeguard transmitted data against snooping efforts. The security of speech is the subject of this study. Speech scrambling, which typically tends to be found in speech samples to remove residual intelligibility of communicated speech, is one of many techniques used to secure speech. The chaotic system has a high level of randomness, making it an excellent choice for generating the scrambling key. Furthermore, orthogonal processing is utilized to diminish the voice signal's intelligibility. To improve the hybridization of the procedures, optimization techniques are applied. An optimization analysis is a good tool that seeks the optimum answer based on a set of fitness criteria among several viable options.

Numerous disciplines have devised a scrambling of speech techniques. To protect audio tracks from snooping or inhibit someone else from understanding reliable data communicated through an insecure channel, many methods designed to satisfy this state rely on one of the popular methods for unexpectedly blending messages and fetching the original data on the other end at the receiver side [1].

The speech scrambling algorithm is commonly implemented as a permutation of the transformation sequence coefficient of the chunks of voice in the time domain, frequency domain, and time-frequency domain [2]. Scrambling technologies have also been utilized to evade eavesdroppers, scrambling technologies have also been utilized to transform the raw voice signal into an unrecognizable form [3].

Chaotic producer

Chaos is a pseudo-random situation that operates in a dynamic and multidimensional system. Nature's chaos is quasi, non-convergent and prone to the beginning states.

Boltzmann's chaotic theory is regarded as a fresh contender for cryptography. This thesis refers to data encryption's properties and benefits [4]. Chaotic systems involve a variety of characteristics, including sensibility that varies depending on the original situation, a continuously broadband spectrum, and ergodicity. The term 'ergodic' contains Greek beginnings and is made up of two phrases: 'ergon,' which means work, and 'odos,' which means road [5]. A decent encryption system must have the main qualities (dispersion and confusion). Chaos and noise are two types of non-regular natural behavior in a real system [4].

After encryption, accurate knowledge of the chaotic system's parameters and initial conditions leads to the retrieval of the raw data. Applying chaos theory in cryptography brings us back to Shannon's universe [5].

Chaotic Maps types

Cryptography makes extensive use of chaotic maps. Several forms of chaos maps are presented in this section, which is employed in the proposed system.

1.Arnold Cats mapping

The Arnold Cats Mapping (ACM) is a simple product of a few chaos concepts [6]. The 2D map provided in Equation (1) is the alteration in this map that is utilized to increase security

$$\begin{pmatrix} a_{x+1} \\ b_{x+1} \end{pmatrix} = \begin{pmatrix} c+1 & x \\ 1 & 1 \end{pmatrix} \begin{pmatrix} a_x \\ b_x \end{pmatrix} (\mathrm{mod}\, S) \qquad (1)$$

Where a_x and b_x are values in a square matrix (S S), s = 1, 2, 3,..., S-1, a_{x+1}, b_{x+1} is the transformation's index mapping, c = 1, 2, 3,...., and S is the square array's length. The suggested system makes use of this map.

2.Fibonacci Sequence

The Fibonacci sequence q is a recurrence of integer numbers that resembles as follows

$$Fib_q = \begin{cases} 0\ if\ q=1 \\ 1,if\ q=2 \\ Fib_{q-1}+Fib_{q+1},\ else \end{cases} \quad (2)$$

The values represented by the created series are: 0, 1,1,2,3,5,8,.... The Fibonacci series is organized into four successive Fibonacci numbers in the 2x2 matrix. The unimodular matrix will be used to scramble other matrices. The Fibonacci Transform in its generalised form is shown in [6]. The mapping of Fibonacci is represented as follows

M:F2-> F2 such that

$$\begin{pmatrix} a' \\ b' \end{pmatrix} = \begin{pmatrix} Fib_i & Fib_{i+1} \\ Fib_{i+2} & Fib_{i+2} \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} (\mathrm{mod}\, S) \quad (3)$$

3.Lucas sequence

The Lucas series is a subset of the Fibonacci sequence, and it is defined as follows:

$$Lu_q = \begin{cases} 2\ if\ q=0 \\ 1,if\ q=1 \\ Lu_{q-1}+Lu_{q-2},\ else \end{cases} \quad (4)$$

The index is q, and the Lucas value is Lu. The following numbers are formed by the series: 2, 1, 3, 4, 7, 11, 18, 29,... [6].

4.**Arnold–Lucas Transform**

The map is made up of the very first row from the Arnold map and the next row from the Lucas maps [7], as seen here:

$$\begin{pmatrix} a' \\ b' \end{pmatrix} = \begin{pmatrix} D_1 & D_2 \\ Lu_1 & Lu_2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} (\mathrm{mod}\, S) \quad (5)$$

Ant Mating Optimization

This technique is the bio inspired approach of real life span of each ant. In the real time during mating the queen ant is selected from the population and the male ant which are in the near location are chosen to mate with the female ant. Hence there occurs the crossover between the queen and male ant and produces the new offspring which are included in the population and the male ant dies during this process. Then the population nurtures the offspring and few female ants provide security and then the life cycle gets continued for the new offspring it produces.

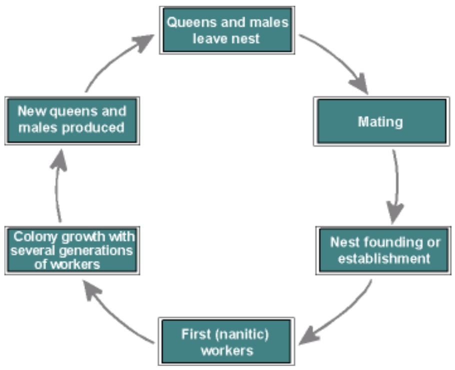

Figure 1 Ants Life cycle

The AMO algorithm begins with initialization of pseudo random allocation in which the ant population consists of both male and female ant. In the whole population female is of 20% to 40% and male is of 60% to 80%. Separate the Ant population as male and queen ant and perform the genetic operations like selection, crossover and mutation to provide the new set of ant solutions.
Then the population is divided into two in the range of 50% and 70 % in population A which mainly consists of female and produced offspring. Female ant nurtures the offspring produced. Population B is separated as 20% to 50% whose objective is to provide protection to the Ant population. Finally these two populations are merged to form the new colony of ant in which certain ants are eliminated based on failing the conditions to meet in the ant colony.

Selection

The selection operation is to choose the male ant for the group of queen ant which is available for mating. Fitness function, probability of choosing ant, Distance function are some of the criteria to be undertaken while selecting the male for the particular queen.

Number of males to be crossover with the particular female and is derived from the random function value and is given as

$$N_c^{qk} = [rand1 \times M_Q] \qquad (6)$$

The minimum distance must be between female and male ant to get crossover hence the **distance between the ants** is deduced as

$$d_{m_i \to q_k} = \sqrt{(X_i^{m_i} - X_j^{q_k})^2} \qquad (7)$$

The **probability of selecting particular male** ant from the population is derived by the following equation

$$P_{m_i} = \frac{F'_{m_i \to q_k}}{\sum_{i=1}^{N_m} F'_{m_i \to q_k}} \quad (8)$$

Degree of pheromone value is obtained from the below result

$$f'_{m_i \to q_k} = \frac{c_{m_i}}{d_{m_i \to q_k}} \quad (9)$$

Based on the probability value the ant is selected for mating with the queen ant will produce the better offspring on mating.

Crossover

Crossover operation is carried out between the male and female ant where the best fit male is calculated based on the fitness value each female ant and mates with ant having the minimum fitness value.The roulette wheel selection method is used to select the male for the queen. The crossover formula is defined as

$$x_n^{larva} = q_k + (\underset{r}{\rightarrow}.\omega.(q_k - m_i)) \quad (10)$$

Mutation

Mutation is the evolutionary process to obtain the efficient optimal solution to the problem; hence mutation is defined as following

$$new.x_b^{larva} = x_b^{larva} + M.\underset{r1}{\mapsto}.x_b^{larva} \quad (11)$$

Measurements for evaluation

The following are the most important factors to consider while considering speech scrambling techniques:

1. The scrambler's strength is that it outputs a scrambled signal with limited resolution.
2. The extent to which encoding or decoding affects the receiver's recovered voice quality.
3. The scrambler's or encryption system's resistance to cryptanalytic attack [8][9]

For calculating residual intelligibility and speech quality, testing employs a variety of metrics. We employ a variety of metrics, such as correlation coefficients (CC) and signal-to-noise ratio (SNR) (SNR). The following points are clarified in greater detail:

1.correlation coefficients (CC)

CC considers quantitative measurements that evaluate an encryption system's quality. The link between two examples of number in the range of +1 to 1 is examined in this study. A correlation score of zero indicates the weakest association between the two data and the attackers' inability to guess the secret key. The following is how CC is calculated:

$$Ei(A)=\frac{1}{D}\sum_{j=1}^{D} A_j \tag{12}$$

$$Di(A)=\frac{1}{D_n}\sum_{j=1}^{Dn}(A_j - Ei(A))^{\wedge 2} \tag{13}$$

$$COVi(A,B)=\frac{1}{D}\sum_{j=1}^{D}(A_j - Ei(A))(B_j - Ei(B_j)) \tag{14}$$

$$R_{AB}=\frac{COVi(A,B)}{\sqrt{Di(A)Di(B)}} \tag{15}$$

2.Signal to noise ratio

SNR evaluates the encrypted signal's accessibility and purity. In general, a low SNR signal suggests a larger noise level than the raw voice signal, while a high SNR signal shows that the decoded signal is of excellent quality [10][11]; SNR is computed as follows

$$SNR=10\log_{10}\frac{\sum_{j=1}^{K} T^2(j)}{\sum_{j=1}^{ks}(T(j)-Di(j))^2} \tag{16}$$

Where Di(j) is the decoded speech value, and T (j) is the actual speech

3.Distance of Log Spectral

Because the perceived sound of a sign is logarithmic in nature, this measure is closely related to self-assessment of voice disparities [12] For the matching frame of the encrypted/restored signal, the logarithmic difference between the Sp (w) original speech signal power spectrum and the power spectrum Sp'(w) can be calculated as follows:

$$VP(X)=\log Sp\,(X)-\log Sp,\,(X), X=0,1,2,\ldots N-1 \tag{17}$$

The dilog among the densities Sp and Sp' is represented as follows

$$Dilog = (Sp, SP,)^{o}=\frac{1}{N}\sum_{j=0}^{N-1}|VP(X)|^{o}\ , x=0,1,2,\ldots N-1 \tag{18}$$

Where o is the distance measurement order

Proposed System

Scrambling and descrambling AMO using chaotic maps are the two phases of the proposed system.AMO standard algorithm has various parameters that must be determined before the algorithm can be used.
Algorithm 1 shows the steps in the AMO process.

1.BEGIN
2. Generate $\mathcal{N}$ pseudorandom solutions;
3. Initialize parameters
4. Calculate the fitness of each ant;
5. **For** k=1: Max Rep {
6. select NQnumber of population as queens and other as a males ($\mathcal{N}\mathcal{N}mm$)
7. %Selecting males for Queen
8.Queens.MaleForQueen = Selection (Queens(i), Males(j),
9. Queens(i). Cost)
10. %randomly selecting queens for each subpopulation
11. Sub_Gen %given subpopulation generation
12. In Sub_Gen do {
13. **For** all Ants in subpopulation do {
14. if rand > $\mathcal{O}\mathcal{O}$ {Queens. Egg = Crossover (Queens, Males);
15. if rand <cmf {Queens. Egg = Mutation (Queens. Egg, mf)}
16. Queens. Egg. Cost = Cost Function (Queens. Egg)}}}
17. % Create New Ant Population
18. if numel (Ant) is greater {
19. Ant=Ant(I(1:(Extinction)));}}
20. Ant = Sort(Ant);
END Algorithm

1) Scrambling Using AMO with Chaotic Maps

The voice vector is fed into two chaotic maps, which are then subjected to the AMO. After N cycles, the best ant is chosen based on its fitness. The ant is a combination of the two input maps. The key used in scrambling is represented by that ant.(ACM and Fibonacci map), (ACM and Lucas map), and (Lucas map and Fibonacci map). are the two maps that can be input into the AMO

Algorithm 2: Algorithm of Scrambling using AMO with Chaotic Maps

Step 1: Transform the voice vector to a two - dimensional array
Step 2: Use Equations (1) and (3) to perform (ACM and Fibonacci map) for (ACM and Lucas map) apply equation 1 and 5 and (Fibonacci map and Lucas map) using equation (3) and (5)
Step 3: Reassemble the two matrices into a 1D array.
Step 4: Use Algorithm 1 to perform AM).
Step 5: As the scrambling key, choose the best solution.

The voice is transformed to a two - dimensional array, then two chaotic maps (ACM and Fibonacci map), (ACM and Lucas map), or (ACM and Lucas map) are created from the states (Fibonacci map and Lucas map).The output is fed into the AMO, with the superior ant acting as the scrambling key.

2) Descrambling using AMO with Chaotic Maps

The original clear speech signal is recovered using the descrambling technique on the second side (the receiver side).The Algorithm 3 explains how to decode using this method.

Algorithm 3: Algorithm of Descrambling using AMO with Chaotic Maps

Step 1: Practice scrambling with the key.

Step 2: Look for and remove any extra zeros at the end of the descrambled vector.

Step 3: Save the resulting voice vector in a legible format.

The scrambling key will be provided surreptitiously to the recipient side. This key will be applied to the scrambled signal, resulting in a descrambled vector that is longer than the plain speech signal's initial length. As a result, the descrambled signal is stripped of the extra zeros at the end of the descrambled vector.

Results and Discussion

In this section, we apply the optimisation process to try to find a better performance, which we decide on based on its outputs when we employ the best hybrid. The optimisation algorithm we utilise is AMO. Two chaotic maps were used to jumble the speech data (ACM and Fibonacci map, ACM and Lucas map, or Fibonacci map and Lucas map). These scrambled vectors were used as the starting point for the PSO algorithm, which updates position X, which represents the scrambled vectors, in each iteration and loops until the stop condition is satisfied. Finally, as scrambled speech data, the best overall anr will be used. Table 1 compares the results of the classic hybrid with the SNR of the three states employed in the study with AMO. In Table 1, The hybrid (ACM, Fibonacci map) determined by AMO gives the good SNR that reached 1.003032, the best value of SNR for (ACM, Lucas map) was 1.00769, and the least value for the state (Fibonacci map, Lucas map) was **1.006371**.

The logarithmic spectral distance for the three phases was also calculated using classical hybrid and AMO. Table 2 compares the results of the AMO with the classical hybrid on the same states and shows the log spectral distance of the three states. Table 2 shows the greatest value of log spectral distance obtained using AMO, which executed the identical states as previously described. The best log spectral distance for the hybrid (ACM and Fibonacci map) performed by AMO was 3.183244, the best log spectral distance for (ACM and Lucas map) was 3.158759
, and the best value for the state (Fibonacci map and Lucas map) was 3.183083

Table 1 SNR for AMO with Chaotic Maps_ Scrambling Process

Traditional Hybrid				**AMO**		
File	ACM, FIB	ACM, Lucas	FIB, Lucas	ACM, FIB	ACM, Lucas	**FIB, Lucas**
1	1.36687	1.30228	1.29318	1.197684	1.2261	**1.18166**
2	1.1056	1.1308	1.05653	1.082679	1.108917	**1.006371**
3	1.2631	1.2369	1.23037	1.212183	1.197917	**1.181614**
4	1.13015	1.14283	1.15204	1.020784	1.009988	**1.078703**
5	1.28547	1.27743	1.28132	1.015213	1.013714	**1.117925**
6	1.26272	1.29434	1.25702	1.177292	1.140729	**1.19985**
7	1.13101	1.32885	1.3083	1.28804	1.282477	**1.233063**
8	1.2701	1.26309	1.20699	1.148985	1.14984	**1.17099**
9	1.0788	1.08548	1.08321	1.003032	1.00769	**1.040931**
10	1.4364	1.4395	1.45554	1.248468	1.286225	**1.278911**

Table 2 Log spectral distance for AMO with Chaotic Maps_ Scrambling Process

Traditional Hybrid				AMO		
File	ACM, FIB	ACM, Lucas	FIB, Lucas	ACM, FIB	ACM, Lucas	**FIB, Lucas**
1	1.68962	1.579991	1.58208	3.183244	3.149824	3.183083
2	1.82278	1.710579	1.69824	3.12805	3.117545	3.123617
3	1.717869	1.60083	1.60332	3.160112	3.136439	3.14446
4	1.719774	1.5606785	1.50457	3.162549	3.157767	2.195471
5	1.67289	1.583095	1.576355	3.163874	3.14775	3.161278
6	1.48281	1.33992	1.35622	3.176346	3.158759	3.174934
7	1.67540	1.59526	1.58957	3.089543	3.071735	3.068247
8	1.661933	1.568762	1.57902	3.123455	3.115704	3.121861
9	1.71327	1.611463	1.62057	3.101375	3.095367	3.111212
10	1.60296	1.500998	1.50526	3.119868	3.104324	3.110998

The correlation in between three states was also assessed using classic hybrid and AMO. Table 3 depicts the correlation between the three states and compares the results obtained using AMO to those obtained using a classical hybrid on the same states.
In Table 3, the best correlation value was obtained by employing AMO to conduct the identical states as previously described. The best correlation for the hybrid (ACM and Fibonacci map) performed by AMO was -0.0218 , the best value of correlation for the (ACM and Lucas map) was -0.01868 , and the greatest value for the state (Fibonacci map and Lucas map) was -0.01837

Table 3 Correlation for AMO with Chaotic maps_ Scrambling Process

Traditional Hybrid				AMO		
File	ACM, FIB	ACM, Lucas	FIB, Lucas	ACM, FIB	ACM, Lucas	FIB, Lucas
1	0.02491	0.01034	0.0082	-0.00401	-0.01092	-0.013
2	0.009298	0.0898	0.00272	-0.01722	-0.01702	0.000982
3	0.0152	0.00882	0.00819	-0.01671	-0.01801	-0.01217
4	-0.00452	−0.00693	0.01111	-0.0021	-0.01712	-0.00209
5	0.005132	−0.0054	0.00418	-0.01899	-0.01501	-0.00164
6	0.0098	0.0822	0.00233	-0.01823	0.060719	-0.0029
7	0.0583	0.0027	0.00236	-0.01968	-0.01868	-0.0028
8	0.0919	0.0756	−0.00394	-0.00315	-0.0029	-0.01705
9	-0.00116	0.0064	0.00161	-0.00218	-0.00609	-0.01637
1	0.01648	−0.03689	0.1933	-0.0218	-0.01813	-0.01837

Tables 4, 5, and 6 explain the metrics utilised in the descrambling process. The quality of the suggested algorithm of the proposed system for extracting actual speech was measured using the three measures derived for descrambling. Table 4 shows that PSO's hybrid (ACM and Fibonacci map) produced the best SNR of 86.66598 while the highest SNR for (ACM and Lucas map) was 86.67012 , and the optimum price value for the state (Fibonacci map and Lucas map) was 86.66562

Table 4 SNR for AMO with Chaotic maps_ de Scrambling Process

Traditional Hybrid				AMO		
File	ACM, FIB	ACM, Lucas	FIB, Lucas	ACM, FIB	ACM, Lucas	**FIB, Lucas**
1	86.521391	86.521091431	86.5416913	86.5397966	86.556342	**86.556721**
2	86.525141	86.503843062	86.5244429760878	86.645982	86.558988	**86.547812**
3	86.465685	86.455385517	86.4759854310635	86.467210	86.490215	**86.489019**
4	86.440680	86.430380308	86.450980221467	86.450911	86.458921	**86.479019**
5	86.465019	86.45471998	86.4753198974726	86.4651	86.47312	**86.490324**
6	86.438686	86.416561718	86.4371616320685	86.4390	86.490381	**86.458321**
7	86.491839	86.481539077	86.5021389912658	86.49219	86.48019	**86.512844**
8	86.469577	86.447448189	86.4568048102694	86.46925	86.45099	**86.456602**
9	86.341139	86.33083980	86.3514397218729	86.345127	86.35980	**86.409353**
10	86.642908	86.618787375	86.63387288439	86.64563	86.65012	**86.645621**

For descrambling, the log spectral distance for the three states was calculated using classic hybrid and AMO. Table 5 displays the log spectral distance of the three phases and compared the AMO results to the classical hybrid results for descrambling the very same states. The optimal value of log spectral distance was the lowest value, as seen in the table.
The optimum values of the log spectral distance was obtained in Table 5 by utilising the AMO to conduct the identical states as previously described. The best log spectral distance was achieved by AMO's hybrid (ACM and Fibonacci map), the best log spectral distance for (ACM, Lucas map) was 0, and the best value for the state was 0.

For descrambling, the correlation between the three phases was examined using classic hybrid and AMO. Table 6 displays the correlation between the three phases and compares the results obtained using the AMO to those obtained using the classic hybrid on the same states, with the best correlation value in the descrambling process being the value closest to 1.
Table 6 shows the best correlation value obtained by applying PSO to conduct the same states as those described earlier. The (ACM and Fibonacci map), (ACM and Lucas map), and (Fibonacci map and Lucas map) hybrids were all 1.

The hybrid using traditional methods and the hybrid using AMO were utilised for comparison between the three states using traditional methods and the three states using AMO for the three measures used for comparison. According to the metrics, the best state in the hybrid utilising PSO was (ACM and Fibonacci map), which reached 15 states in scrambling and 22 states in descrambling.

Table 5 Spectral distance for AMO with Chaotic maps_ de Scrambling Process

Traditional				AMO		
File	ACM, FIB	ACM, Local	FIB, Lucas	ACM, FIB	ACM, Lucas	**FIB, Lucas**
1	0	0.00000032	0.00000064	0	0.00000009	**0.00000001**
2	0	0.00000055	0.00000109	0	0.00000010	**0.00000101**
3	0.00000002	0.00000653	0.00001244	0	0	**0.00000101**
4	0	0.00000036	0.0000072	0	0	**0**
5	0	0.00000031	0.00000062	0	0.00000005	**0**
6	0	0.00000004	0.00000089	0.00000038	0	**0**
7	0	0.00000605	0.00001182	0	0	**0**
8	0.00000001	0.0000001	0.00000095	0	0.00000007	**0**
9	0	0.00000018	0.00000177	0	0.00000002	**0**
10	0.000000061	0.00000013	0.00000063	0	0	**0**

Table 6 correlation for AMO with Chaotic maps_ deScrambling Process

Traditional Hybrid				**AMO**		
File	ACM, FIB	ACM, Lucas	FIB, Lucas	ACM, FIB	ACM, Lucas	**FIB, Lucas**
1	1	1	1	1	1	1
2	1	1	1	1	1	1
3	1	1	1	1	1	1
4	1	1	1	1	1	1
5	1	1	1	1	1	1
6	1	1	1	1	1	1
7	1	1	1	1	1	1
8	1	1	1	1	1	1
9	1	1	1	1	1	1
10	1	1	1	1	1	1

Conclusions

The application of the AMO to improve the performance of the methods employed in voice scrambling is the subject of our proposed system. The optimal hybrid in AMO with chaotic maps is offered by ACM and Fibonacci map, according to our findings.

References

1. Sathiyamurthi, P., & Ramakrishnan, S. (2017). Speech encryption using chaotic shift keying for secured speech communication. EURASIP Journal on Audio, Speech, and Music Processing, 2017(1), 1-11.
2. Duta, C.L., Gheorghe, L., & Tapus, N. (2016). Performance Analysis of Real Time Implementations of Voice Encryption Algorithms using Blackfin Processors. In International Conference on Information Systems Security and Privacy, 2, 157-166.
3. Sadkhan, S., & Abbas, N. (2012). Speech scrambling based on wavelet transform. Advances in wavelet theory and their applications in engineering physics and technology, 41-58.
4. Al Saad, S.N., & Hato, E. (2014). A speech encryption based on chaotic maps. International Journal of Computer Applications, 93(4), 19-28.
5. Ismael, H.A. (2017). A proposed Speech Scrambling Based on Multi Chaotic Maps as Key Generators (Doctoral dissertation, Master dissertation, University of Babylon College of Information Technology Department of Software, Iraq, 2107).
6. Mishra, M., Mishra, P., Adhikary, M.C., & Kumar, S. (2012). Image encryption using Fibonacci-Lucas transformation. International Journal on Cryptography and Information Security (IJCIS), 2(3).
7. Abbas, N.A., & Razaq, Z.H. (2018). Speech Scrambling Based on Arnold-Lucas Mapping. In International Conference on Advanced Science and Engineering (ICOASE), 290-295.

8. Weise, T. (2009). Global optimization algorithms-theory and application. Self-Published Thomas Weise.
9. Munakata, T. (2008). Fundamentals of the new artificial intelligence: neural, evolutionary, fuzzy and more. Springer Science & Business Media.
10. Sohail, M.S., Saeed, M.O.B., Rizvi, S.Z., Shoaib, M., & Sheikh, A.U.H. (2014). Low-complexity particle swarm optimization for time-critical applications.
11. Ar Xiv preprint arXiv:1401.0546.
12. Hassan, R., Cohanim, B., De Weck, O., & Venter, G. (2005). A comparison of particle swarm optimization and the genetic algorithm. In 46th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference.
13. Sun, J., Wu, X., Palade, V., Fang, W., Lai, C. H., & Xu, W. (2012). Convergence analysis and improvements of quantum-behaved particle swarm optimization. Information Sciences, 193, 81-103.
14. Bai, Q. (2010). Analysis of particle swarm optimization algorithm. Computer and information science, 3(1), 180-184.
15. Kohad, H., Ingle, V.R., & Gaikwad, M.A. (2012). An overview of speech encryption techniques. International journal of Engineering research and development, 3(4), 29-32.
16. Ghassan, M.H. (2014). Speech Scrambling Using Multi-Stage Permutation with Filter Output. Iraqi Journal of Information Technology, 6(2), 90-101.
17. Farsana, F.J., & Gopakumar, K. (2016). A novel approach for speech encryption: Zaslavsky map as pseudo random number generator. Procedia computer science, 93, 816-823.
18. Mosa, E., Messiha, N.W., Zahran, O., & Abd El-Samie, F.E. (2011). Chaotic encryption of speech signals. International Journal of Speech Technology, 14(4), 285-296.
19. Doukas, N., & Karadimas, N.V. (2008). A blind source separation based cryptography scheme for mobile military communication applications. WSEAS Transactions on Communications, 7(12), 1235-1245.
20. Rachman, Y.B., Mutiarani, H., & Putri, D.A. (2018). Content analysis of Indonesian academic libraries' use of Instagram. Webology, 15(2), 27-37.