Voice Scrambling Algorithm based on 3D Chaotic Map System (VSA3DCS) to Encrypt Audio Files

Osama M. Abu Zaid
Lecturer in Computer Science Dept., Faculty of Computers &
Information, Kafrelsheikh Univ., Egypt

***Abstract*—Here, a proposed voice scrambling algorithm established on one of two 3D chaotic maps systems (VSA3DCS) will be presented, discussed, and applied on audio signals file. The two 3D chaotic map systems in which any one of them is used to build VSA3DCS are Chen's chaotic map system and Lorenz chaotic map system. Also Arnold cat map-based scrambling algorithm will be applied on the same sample of audio signals. These Scrambling algorithms are used to encrypt the audio files by shuffling the positions of signals at different conditions with the audio file as one block or two blocks. Amplitude values of audio signals with signals' time are registered and plotted for original file versus encrypted files which are produced from applying VSA3DCS using Chen's, VSA3DCS using Lorenz, and Arnold-based algorithm. The spectrogram frequencies of audio signals with signals' time are plotted for original file versus encrypted files for all algorithms. Also, the histogram of the original file and encrypted audio signals are registered and plotted. The comparative analysis is presented by using some measuring factors for both of encryption and decryption processes, such as; the time of encryption and decryption, Correlation Coefficient of original and encrypted signals between the samples, the Spectral Distortion (SD) measure, Log-Likelihood Ratio (LLR) measure, and key sensitivity measuring factor. The results of several experimental and comparative analyses will show that the VSA3DCS algorithm using Chen's or Lorenz is a good algorithm to provide an effective and safe solution to voice signal encryption, and also VSA3DCS algorithm better than Arnold-based algorithm in all results with all cases.**

***Keywords*—Lorenz chaotic map; Chen's chaotic map; Arnold cat map; scrambling algorithms; audio encryption**

I. Introduction

Now we are living amid a digital revolution that needs safe multimedia transmission. Visual encryption is essential when transmitting audio over communication networks to protect them from reading, altering their content, inserting false information, or deleting a portion of their content [1,2].

Multimedia encryption has recently become one of the key problems of great concern. It offers greater protections for the content, which may involve some private issues or save copyrights from being changed or violated [3]. Any cryptography process requires a simple algorithm with tiny processing time and high performance to protect the information. Besides that, it has a strong immune system against any external issue including noise and interference that can be faced in the channels of communication [3].

A chaotic map is a suitable solution for both issues (tiny processing time and high performance). As with other methods of encryption such as AES and DES, which have large processing then a long time, the chaotic map has a fair time to fit these tasks [3,4].

As it is possible any unauthorized person can receive the transmitted data with the simplest receivers, the security of audio conversations has recently become a crucial issue because of the successful development of crypt-analysis activities [1,5]. Chaos-based encryption mechanisms are considered to be ideal for practical use because they provide an honest combination of speed, high security, complexity [4,6,7].

In this work, it tries to solve these two challenges by producing a proposed voice scrambling algorithm (VSA3DCS) based on a 3D chaotic map system (Lorenz map system or Chen's map system). The VSA3DCS algorithm is compared with one of the 2D chaotic maps (Arnold Cat map) which used to permute the elements in the multimedia file (image or audio). Also, in this work, several metrics are evaluated to accomplish comparative analysis.

This research paper is arranged as follows: Section II will present the related work, motivation, and contribution. Section III will present the chaotic maps which are used in our work. Section IV will present the steps of the proposed algorithm VSA3DCS. Section V will present applying all algorithms on the same audio signals file. Section VI will discuss experiential results and comparative analysis. Section VII will discuss the conclusion. In the final, there are references which are being used.

II. Related Work, Motivation And Contribution

Most of the research papers in the cryptography field use many chaotic maps systems of various dimensions or any other techniques in image encryption.

Many of the research papers apply only one-dimensional or two-dimensional chaotic maps systems in audio encryption, whereas most of these papers are is to produce an algorithm for changing in the values of signals (substitution encryption). In [3], E. Mosa et al. implemented a voice encryption method based on permutation of voice segments using a 2D chaotic map (Baker map) and substitution using masks in time and transform domains. In [8], Arnold cat map was applied by Mahmoud F. Abd Elzaher and others to permute voice samples, then either Henon or modified Henon or Unified or Lorenz chaotic systems were applied to produce the mask key and thus replace the permuted samples. In previous research for me, it is now under review for publication in another valued journal, I applied both systems of 2D chaotic maps (Arnold Cat map and Baker map) which used in the permutation of locations for the elements of the audio signals file. Comparative analyses were made for the results showed that Arnold's application was the least in time of encryption/decryption and the best in performance in most cases.

So, here my research paper introduces a proposed multi-step voice scrambling algorithm that is developed using any one of two well-known types of 3D chaotic maps systems which are strength and sophistication in their use of cryptography operations; they are (Lorenz and Chen's). the proposed algorithm for encoding audio signals in a way that alters and confuses the locations of signals only (transposition encryption) without changing their values, and it is based on either of the two chaotic systems (Lorenz or Chen's) as will be evident from the part that explains the steps of that algorithm and as will be clear from A flow diagram that shows the details of the algorithm. The proposed algorithm is compared with the Arnold-based algorithm.

III. The Chaotic Maps Systems

In this section, a concise description is provided about the two 3D chaotic maps systems used to construct the VSA3DCS algorithm. Also, Arnold cat map will be discussed here.

A. Lorenz Chaotic Map System

The Lorenz Chaotic map system is a three-equation scheme. The Lorenz system equations are defined as in Formula (1) [6,7,8,9,10].

$$\begin{cases} x = \sigma(y_0 - x_0) \\ y = rx_0 - y_0 - x_0 z_0 \\ z = x_0 y_0 - b z_0 \end{cases} \tag{1}$$

Where *σ, r, b* are the parameters of this chaotic system. The system displays unpredictable behavior when $\sigma = 10$, $r > 24.74$ and $b = 8/3$. The initial state values x_0, y_0, and z_0 act as the keys to the diffusion. A very good result for Lorenz chaotic map with the parameters $\sigma = 10$, $r = 28$, $b = 8/3$, and $h = 0.1$, the initial values $x_0 = 10$, $y_0 = 20$, $z_0 = 30$, where, *h* is the sequence step. The Lorenz system attractor is illustrated in Fig. 1.

B. Chen's Chaotic Map System

As one of the 3-D chaotic map systems defined by formula (2), Chen's chaotic map system is essential as a collection of the three differential equations of Chen's chaotic map system [7,10,11,12,13].

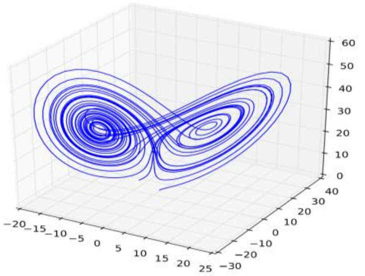


Fig 1. Chaotic Behavior of the Lorenz System.

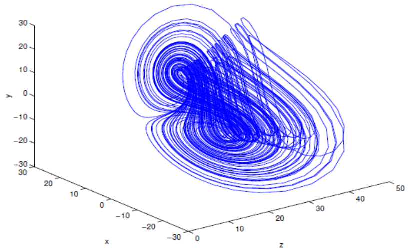


Fig 2. Chaotic Behavior of Chen's System.

$$\begin{cases} x = a(y_0 - x_0) \\ y = (c - a)x_0 - x_0 z_0 + c y_0 \\ z = x_0 y_0 - b z_0 \end{cases} \tag{2}$$

Where, $a > 0$, $b > 0$, and c so $(2c > a)$ are system parameters. Chen's chaotic map system is chaotic with the parameters' values; $a = 35$, $b = 3$, and $c \in [20, 28.4]$.

There is also another parameter (*h*), such that *h* is the increasing step value of x_0, y_0, and z_0 for each round, that is, $x_0 = x_0 + h$, $y_0 = y_0 + h$, and $z_0 = z_0 + h$. If $a = 35$, $b = 3$, and $c = 28$; as shown in Fig. 2, it has a chaotic attractor. A very good result for this chaotic map with the parameters $a = 35$, $b = 3$, $c = 28$, and $h = 0.05555$, the initial values $x_0 = 0$, $y_0 = 1$, $z_0 = 0$, where *h* is the sequence step.

C. Arnold Cat Map

The Arnold Cat map is a chaotic map which is invertible in 2-D. For shuffling the pixel positions of the plain image or positions of signals in an audio file, we choose Arnold cat map method [6,14,15].

Without lack of generality, we assess the dimension of the multimedia file as *N*×*N* (it may be *N*×*M*). Arnold map method as shown in Formula (3) [6,14,16,17]:

$$\begin{aligned} \begin{bmatrix} x_{m+1} \\ y_{m+1} \end{bmatrix} &= A \begin{bmatrix} x_m \\ y_m \end{bmatrix} (mod\ N) \\ &= \begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \end{bmatrix} (mod\ N) \\ &= \begin{bmatrix} x_m + p y_m \\ q x_m + (pq+1) y_m \end{bmatrix} mod\ N \end{aligned} \tag{3}$$

Where *p* and *q* bear positive quantities because $det(A)=1$ from $(pq+1)$-pq. The (x_{m+1}, y_{m+1}), when Arnold map is applied once, is the novel position of the original pixel position (x_m, y_m). Where that $m=0,1,2,3,$ .... There are positive integer *T* at repeated *R* times, where at *T*, $(x_{m+1}, y_{m+1}) = (x_m, y_m)$.

IV. VSA3DCS Algorithm

In this part of the paper, the proposed voice scrambling Algorithm (VSA3DCS) based on one of two 3D Chaotic Maps Systems (Chen's or Lorenz) is presented. VSA3DCS consists of a scrambling procedure to produce a shuffled audio file and return-scrambling procedure to reproduce the original audio file. The scrambling algorithm VSA3DCS is designed to shuffle the positions of signals of an audio file. VSA3DCS consists of seven steps of operations as following, and its Data-Flow diagram will be illustrated in Fig. 3:

Step 1: Obtain the *au* vector (1D matrix) of the audio signals file *m*×*1*. The length of *au* is *L* which is equal to *m.*

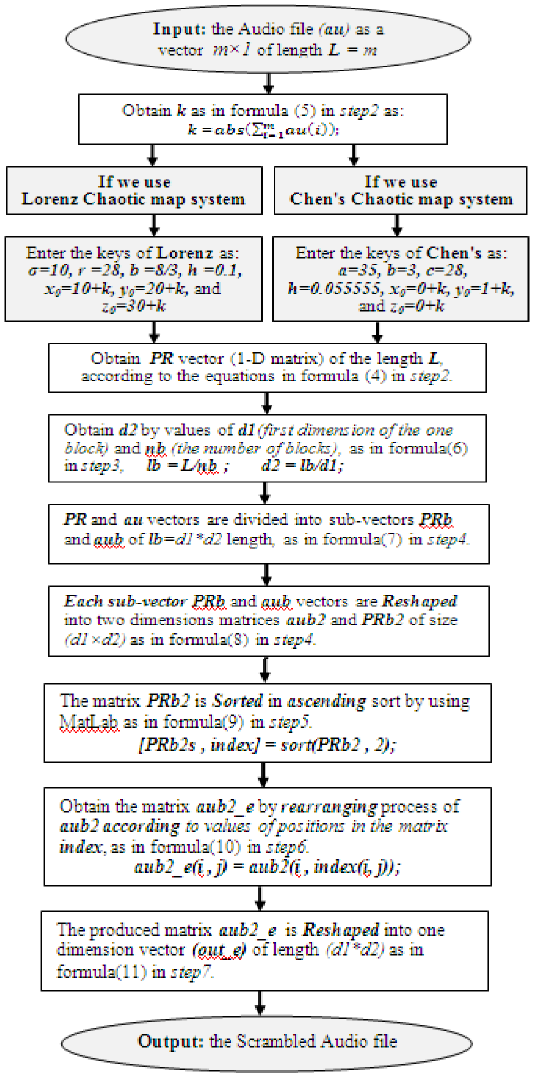

Input: the Audio file (au) as a vector m×1 of length L = m
Obtain k as in formula (5) in step2 as: k = abs(∑ᵐᵢ₌₁ au(i));
If we use Lorenz Chaotic map system
If we use Chen's Chaotic map system
Enter the keys of Lorenz as: σ=10, r =28, b =8/3, h =0.1, x₀=10+k, y₀=20+k, and z₀=30+k
Enter the keys of Chen's as: a=35, b=3, c=28, h=0.055555, x₀=0+k, y₀=1+k, and z₀=0+k
Obtain PR vector (1-D matrix) of the length L, according to the equations in formula (4) in step2.
Obtain d2 by values of d1 (first dimension of the one block) and nb (the number of blocks), as in formula(6) in step3, lb = L/nb ; d2 = lb/d1;
PR and au vectors are divided into sub-vectors PRb and aub of lb=d1*d2 length, as in formula(7) in step4.
Each sub-vector PRb and aub vectors are Reshaped into two dimensions matrices aub2 and PRb2 of size (d1×d2) as in formula(8) in step4.
The matrix PRb2 is Sorted in ascending sort by using MatLab as in formula(9) in step5.
[PRb2s , index] = sort(PRb2 , 2);
Obtain the matrix aub2_e by rearranging process of aub2 according to values of positions in the matrix index, as in formula(10) in step6.
aub2_e(i , j) = aub2(i , index(i, j));
The produced matrix aub2_e is Reshaped into one dimension vector (out_e) of length (d1*d2) as in formula(11) in step7.
Output: the Scrambled Audio file

Fig 3. The Data-Flow Diagram for the VSA3DCS Algorithm.

Step 2: Obtain the *PR* vector as in formula (4), which is vector of numbers with the same length (*L*) of *au*, and generated either by Chen's system at *a*=35, *b*=3,*c*=28, the initial values x_0 =0+*k*, y_0 =1+*k*, z_0 =0+*k*, and *h* =0.055555, or by Lorenz system at σ=10, *r* =28, *b* =8/3, the initial values x_0 =10+*k*, y_0 =20+*k*, z_0 =30+*k*, and *h* =0.1.

$$PR(i) = mod(floor(x),256);$$

$$PR(i+1) = mod(floor(y),256); \tag{4}$$

$$PR(i+2) = mod(floor(z),256);$$

where *i* is from *1* to *L* with increasing step equal 3 for every round in the loop. The *x*, *y*, and *z* values are derived either from the three Lorenz system equations in formula(1) or from the three Chen's system equations in formula(2). *k* is obtained by formula(5), in which the keys in the proposed algorithm are modified.

$$k = abs(\sum_{i=1}^{m} au(i)); \tag{5}$$

Step 3: Obtain *d2* the length of the second dimension of the one block from audio signals' file by using the first dimension (*d1*) for the one block, and the number of blocks (*nb*) which are needed to divide the *au* vector using its length (*L*); this as in formula (6).

$$lb = L/nb \; ; \qquad d2 = lb/d1; \tag{6}$$

Step 4: By the two dimensions *d1*, and *d2*, both *PR* and *au* vectors are divided into sub-vectors of *lb=d1*d2* length, as in formula (7) to produce *PRb* and *aub* sub-vectors over the loop.

$$PRb = PR(1,i{:}i+lb-1); \quad aub = au(1,i{:}i+lb-1); \tag{7}$$

where *i* is from *1* to *L* with increasing step equal *lb* for every round in the loop. Also, each of both vectors *PRb* and *aub* is reshaped by MatLab into two dimensions matrices *aub2* and *PRb2* of size *(d1×d2)*, as in formula (8).

$$aub2 = reshape(aub,d1,d2); \tag{8}$$

$$PRb2 = reshape(PRb,d1,d2);$$

Step 5: Inside the previous loop, the matrix *PRb2* is sorted in *ascending* sort by using MatLab. The Matrix *PRb2s* is produced from this sorting process, as in formula (9).

$$[PRb2s\; ,\; index] = sort(PRb2\; ,\; 2); \tag{9}$$

Also, it returns the matrix of indices *index*, where size(*index*)=size(*PRb2*). For example, if $PRb2 = \begin{bmatrix} 3 & 7 & 0 & 5 \\ 0 & 4 & 5 & 2 \end{bmatrix}$, then *[PRb2s, index] = sort(A,2)* produces the following:

$$PRb2s = \begin{bmatrix} 0 & 3 & 5 & 7 \\ 0 & 2 & 4 & 5 \end{bmatrix}, \text{ and } index = \begin{bmatrix} 3 & 1 & 4 & 2 \\ 1 & 4 & 2 & 3 \end{bmatrix}$$

Step 6: The reshaped matrix *aub2* are rearranged according to the position of *PRb2* in *PRb2s*, i.e., according to values of positions in the matrix *index*, as in formula (10).

$$aub2\_e(i\; ,\; j) = aub2(i\; ,\; index(i,\; j)); \tag{10}$$

For example, let's suppose $aub2 = \begin{bmatrix} 115 & 30 & 50 & 110 \\ 30 & 45 & 65 & 120 \end{bmatrix}$,

Then formula (10) produces the following:

$$aub2\_e = \begin{bmatrix} 50 & 115 & 110 & 30 \\ 30 & 120 & 45 & 65 \end{bmatrix},$$

Note that, at decryption process, we obtain decrypted matrix *(aub2_d)* by using backward process of formula (10) as; *aub2_d(i , index(i, j)) = aub2_e (i , j);*

Step 7: At the end of every round of the loop, the matrix *aub2_e* is reshaped by MatLab into one dimensions matrix (vector) of length *(d1*d2)*, as in formula (11).

$$out\_e = reshape(aub2\_e,1,\; d1*d2); \tag{11}$$

V. Applying VSA3DCS and Arnold on Audio File

In this section, the debate and results of applying the VSA3DCS algorithm and Arnold-based algorithm on audio signals are presented. Original audio signals' file of a conversation between two persons in the time domain (*TD*) which its patterns illustrated in Fig. 4. The length (*L*) of this vector of audio signals for this file is equal to 60416.

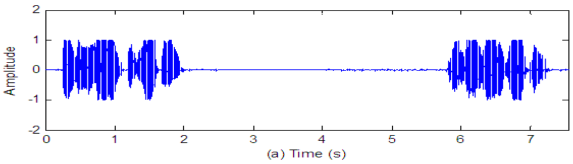


Fig 4. Original Audio Signals' Patterns in the Time Domain.

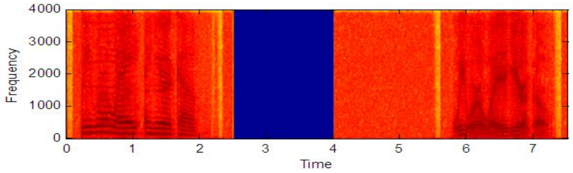


Fig 5. Spectrograms of the Audio Signals for the Original Audio.

The VSA3DCS algorithm is applied on the original audio file either based on Chen's system at *a*=35, *b*=3, *c*=28, the initial values x_0=0+*k*, y_0=1+*k*, z_0 =0+*k*, and *h* =0.055555, or based on Lorenz system at σ=10, *r* =28, *b* =8/3, the initial value x_0=10+*k*, y_0=20+*k*, z_0=30+*k*, and *h* =0.1, whereas *k* is obtained by formula (5). The Arnold-based algorithm is applied to the same original audio file with choice *p=1*, *q=1*, and *R=1*.

All algorithms are applied on the audio file with the first dimension for each one block *d1 = 4, 8, 16, or 32*, and the number of blocks *nb = 1block, or 2blocks*. And the second dimension of each one block *d2* is obtained by formula (6).

Fig. 5 illustrates the spectrograms of the signals of the original audio file, which illustrated in Fig. 4.

A. *Scrambled Audio Signals' Patterns*

Fig. 6 illustrates scrambled audio signals' patterns for the Arnold-based algorithm and the VSA3DCS algorithm based on both systems (Lorenz and Chen's) at the case of *d1=4*, *nb=1*, which led to *d2=15104*, whereas, Fig. 6(a) shows the result for Arnold, Fig. 6(b) shows the result of VSA3DCS with Lorenz, and Fig. 6(c) shows the result of VSA3DCS with Chen's.

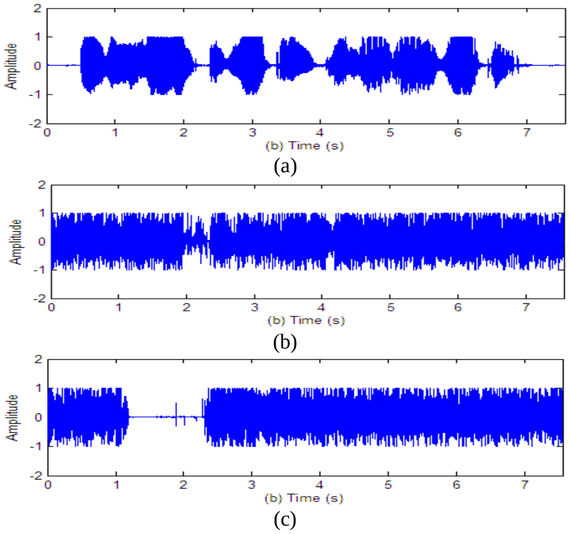


Fig 6. Scrambled Audio Signals' Patterns in TD at *d1=4*, *nb=1*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.

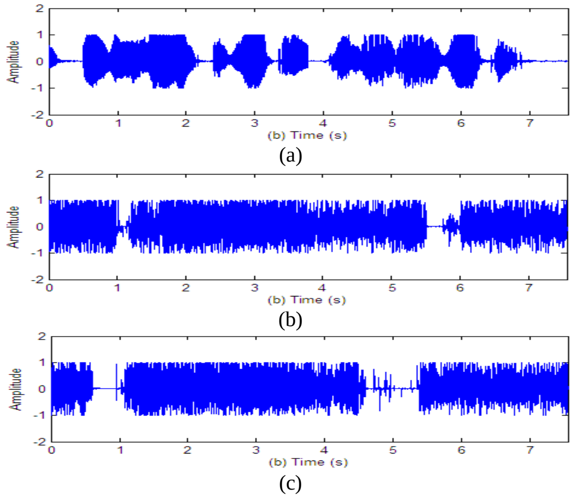


Fig 7. Scrambled Audio Signals' Patterns in TD at *d1=4*, *nb=2*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.

Fig. 7 illustrates scrambled audio signals' patterns for the Arnold-based algorithm and the VSA3DCS algorithm with both chaotic systems at the case of *d1=4*, *nb=2*, which led to *d2=7552*, whereas, Fig. 7(a) shows the result for Arnold, Fig. 7(b) shows the result of VSA3DCS with Lorenz, and Fig. 7(c) shows the result of VSA3DCS with Chen's.

Fig. 8 illustrates scrambled audio signals' patterns for the Arnold-based algorithm and the VSA3DCS algorithm with both chaotic systems at the case of *d1=8*, *nb=1*, which led to *d2=7552*, whereas, Fig. 8(a) shows the result for Arnold, Fig. 8(b) shows the result of VSA3DCS with Lorenz, and Fig. 8(c) shows the result of VSA3DCS with Chen's.

Fig. 9 illustrates scrambled audio signals' patterns for the Arnold-based algorithm and the VSA3DCS algorithm with both chaotic systems at the case of *d1=8*, *nb=2*, which led to *d2=3776*, whereas, Fig. 9(a) shows the result for Arnold, Fig. 9(b) shows the result of VSA3DCS with Lorenz, and Fig. 9(c) shows the result of VSA3DCS with Chen's.

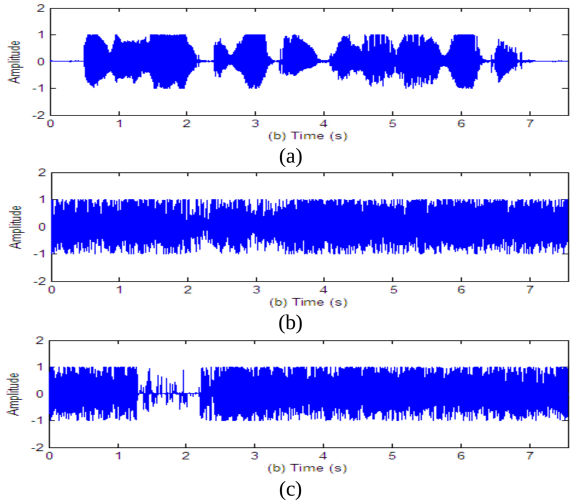
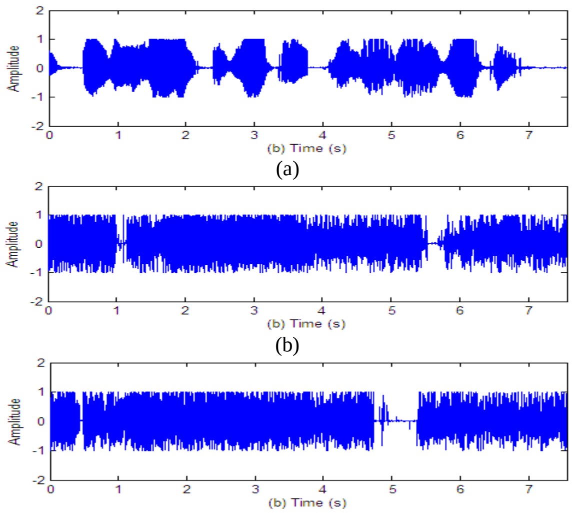


Fig 8. Scrambled Audio Signals' Patterns in TD at *d1=8*, *nb=1*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.

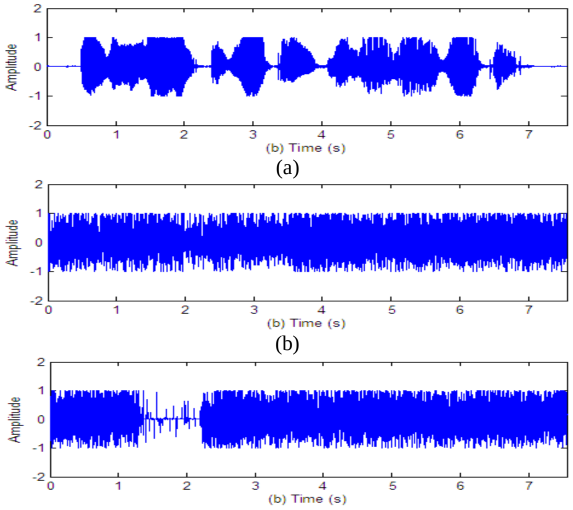


(a)

(b)

(c)

Fig 9. Scrambled Audio Signals' Patterns in TD at *d1=8*, *nb=2*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.



(a)

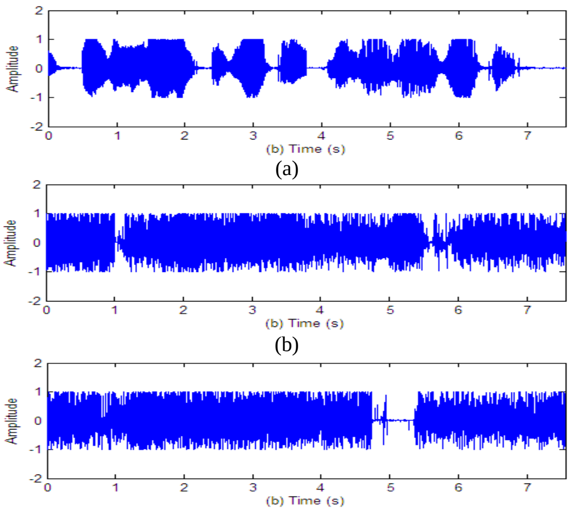
(b)

(c)

Fig 10. Scrambled Audio Signals' Patterns in TD at *d1=16*, *nb=1*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.

Fig. 10 illustrates scrambled audio signals' patterns for the Arnold-based algorithm and the VSA3DCS algorithm with both chaotic systems at the case of d1=16, nb=1, which led to d2=3776, whereas, Fig. 10(a) shows the result for Arnold, Fig. 10(b) shows the result of VSA3DCS with Lorenz, and Fig. 10(c) shows the result of VSA3DCS with Chen's.

Fig. 11 illustrates scrambled audio signals' patterns for the Arnold-based algorithm and the VSA3DCS algorithm with both chaotic systems at the case of d1=16, nb=2, which led to d2=1888, whereas, Fig. 11(a) shows the result for Arnold, Fig. 11(b) shows the result of VSA3DCS with Lorenz, and Fig. 11(c) shows the result of VSA3DCS with Chen's.

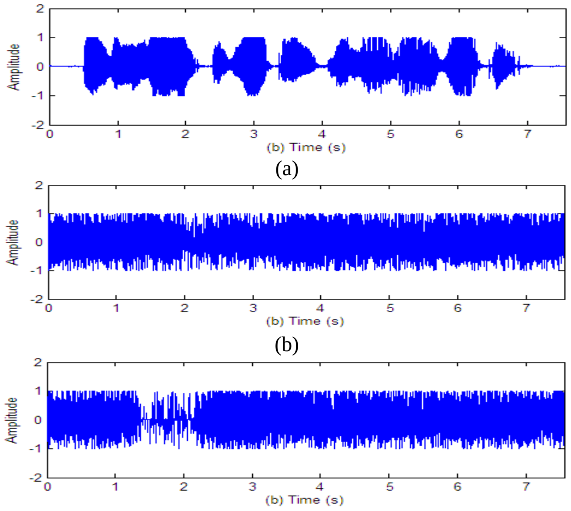


(a)

(b)

(c)

Fig 11. Scrambled Audio Signals' Patterns in TD at *d1=16*, *nb=2*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.



(a)

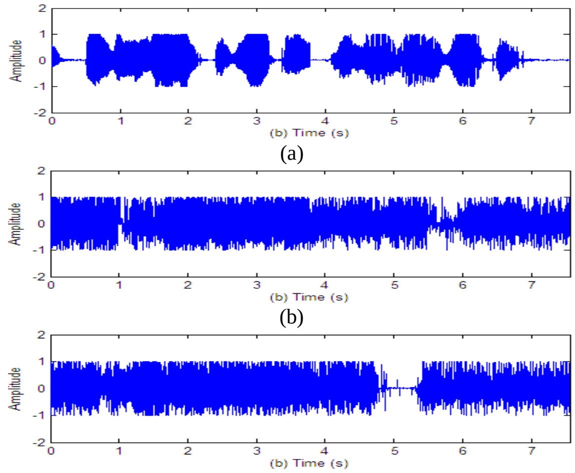
(b)

(c)

Fig 12. Scrambled Audio Signals' Patterns in TD at *d1=32*, *nb=1*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.

Fig. 12 illustrates scrambled audio signals' patterns for the Arnold-based algorithm and the VSA3DCS algorithm with both chaotic systems at the case of *d1=32*, *nb=1*, which led to *d2=1888*, whereas, Fig. 12(a) shows the result for Arnold, Fig. 12(b) shows the result of VSA3DCS with Lorenz, and Fig. 12(c) shows the result of VSA3DCS with Chen's.

Fig. 13 illustrates scrambled audio signals' patterns for the Arnold-based algorithm and the VSA3DCS algorithm with both chaotic systems at the case of *d1=32*, *nb=2*, which led to *d2=944*, whereas, Fig. 13(a) shows the result for Arnold, Fig. 13(b) shows the result of VSA3DCS with Lorenz, and Fig. 13(c) shows the result of VSA3DCS with Chen's.



(a)

(b)

(c)

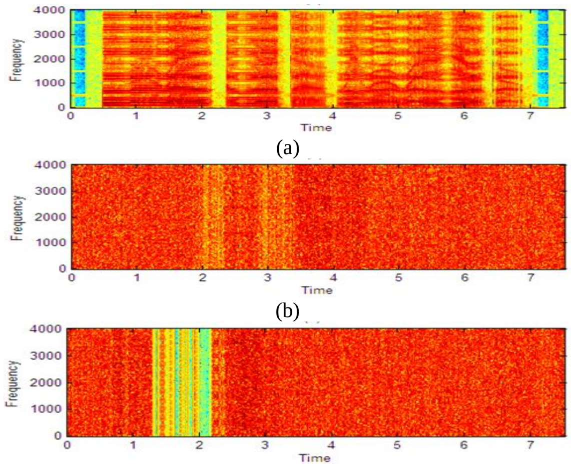
Fig 13. Scrambled Audio Signals' Patterns in TD at *d1=32*, *nb=2*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.

All Fig. 6 to 13, illustrate the results of scrambled audio by VSA3DCS algorithm with both Lorenz and Chen's are completely different than the original audio, and these results of applying the VSA3DCS algorithm are better than the results of applying Arnold-based algorithm at all cases of *d1*and *nb*.

B. *Spectrogram*

A spectrogram reflects a visual representation of the frequency spectrum of a signal, as it varies over time. Generally a spectrogram is represented as an image with the intensity indicated by varying color or brightness.

Some of the results of applying all algorithms are presented, whereas, Fig. 14 illustrates Spectrogram of scrambled audio Signals for the Arnold-based algorithm and the VSA3DCS algorithm based on both systems (Lorenz and Chen's) at the case of *d1=8*, *nb=1*, Fig. 14(a) shows the result for Arnold, Fig. 14(b) shows the result of VSA3DCS with Lorenz, and Fig. 14(c) shows the result of VSA3DCS with Chen's.



(a)

(b)

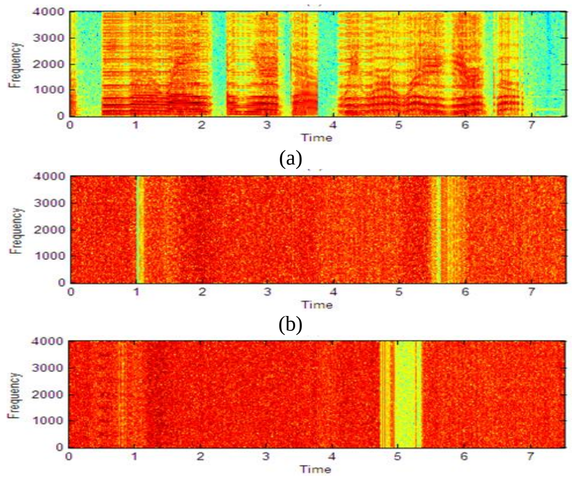
(c)

Fig 14. Spectrogram for Scrambled Audio in TD at *d1=8*, *nb=1*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.

Fig. 15 illustrates Spectrogram of scrambled audio Signals for the Arnold-based algorithm and the VSA3DCS algorithm based on both systems (Lorenz and Chen's) at the case of *d1=16*, *nb=2*, Fig. 15(a) shows the result for Arnold, Fig. 15(b) shows the result of VSA3DCS with Lorenz, and Fig. 15(c) shows the result of VSA3DCS with Chen's.

Fig. 16 illustrates Spectrogram of scrambled audio Signals for the Arnold-based algorithm and the VSA3DCS algorithm based on both systems (Lorenz and Chen's) at the case of d1=32, nb=1, Fig. 16(a) shows the result for Arnold, Fig. 16(b) shows the result of VSA3DCS with Lorenz, and Fig. 16(c) shows the result of VSA3DCS with Chen's.

Fig. 14 to 16 illustrate Spectrogram of scrambled audio with all algorithms is completely different from than spectrogram of the original audio at all cases, and the result of applying the VSA3DCS algorithm with both Lorenz and Chen's is better than the result of applying Arnold-based algorithm at most cases of *d1* and *nb*.

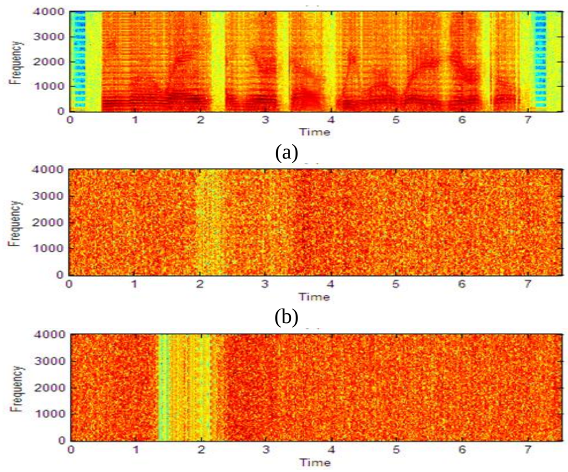


(a)

(b)

(c)

Fig 15. Spectrogram for Scrambled Audio in TD at *d1=16*, *nb=2*. (a) By Arnold-based algorithm. (b) By VSA3DCS algorithm with Lorenz. (c) By VSA3DCS algorithm with Chen's.



(a)

(b)

(c)

Fig 16. Spectrogram for Scrambled audio in TD at *d1=32*, *nb=1*. (a) By Arnold-based Algorithm. (b) By VSA3DCS Algorithm with Lorenz. (c) By VSA3DCS Algorithm with Chen's.

C. Histogram

For continuous data a histogram is used where the bins reflect data ranges. Also, a histogram is an approximate representation of the numerical or categorical data distribution.

Since all algorithms are used for the process of encryption by Scrambling (shuffling of signals' locations) for audio signals, so the histogram of the scrambled audio signals for all cases completely matched to the histogram of the original audio signals which illustrated in Fig. 17.

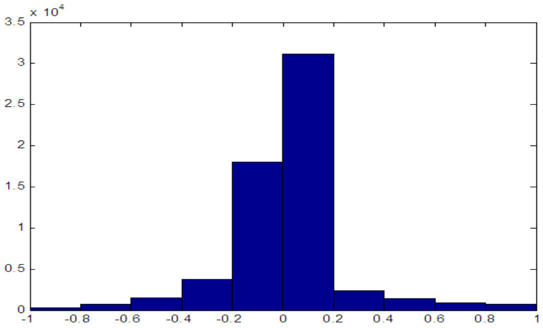


Fig 17. Histogram of the Audio Signals for Original Audio and Scrambled Audio Signals by applying all Algorithms at all Cases.

In the decryption process with all algorithms, all results and plots of audio signals' patterns, spectrogram, and histogram for decrypted audio signals are matched to all plots of the original audio signals illustrated in Fig. 4, 5, and 17, respectively. This indicates that the decryption process is equally successful and efficient with applying all algorithms in all cases.

VI. EXPERIENTIAL RESULTS AND COMPARITIVE ANALYSIS

Here we present experiential findings and comparative analysis using some of several experiential and statistical analyzes for both encryption and decryption procedures, such as encryption and decryption time, correlation coefficient (CC) of evident and encrypted signals between samples, measurement of spectral distortion (SD), measurement of log-likelihood ratio (LLR), and measurement of key sensitivity.

A. Encryption and Decryption Time

In this analysis, for applying all algorithms on the original audio signals file at all cases for both *d1*(*4, 8,* 16, and *32*) and *nb* (*1*, and *2*), the execution time of encryption and decryption has been calculated by seconds.

TABLE I. ENCRYPTION TIME IN SEC. FOR ALL ALGORITHMS

Algorithms	***nb***	**Results of Encryption Time in *Sec.* for all algorithms at all cases of *d1* and *nb***			
		d1=4	***d1=8***	***d1=16***	***d1=32***
Arnold-based Algorithm	***nb=1***	4.3000	1.1070	0.3120	0.1090
	nb=2	2.1840	0.5770	0.1870	0.0690
VSA3DCS with Lorenz	***nb=1***	2.1990	0.5930	0.1870	0.0930
	nb=2	1.1390	0.3280	0.1190	0.0670
VSA3DCS with Chen's	***nb=1***	2.1840	0.5930	0.1870	0.0940
	nb=2	1.1230	0.3280	0.1240	0.0670

Table I, shows the execution time of encryption procedure for applying all algorithms in all cases of *d1* and *nb*. Also, Table II, shows the execution time of decryption procedure for applying all algorithms in all cases of *d1* and *nb*.

Fig. 18 shows the plot for the results of encryption time of applying all algorithms in all cases of *d1* and *nb*. Also, Fig. 19 shows the plot for the results of decryption time of applying all algorithms in all cases of *d1* and *nb*.

TABLE II. DECRYPTION TIME IN SEC. FOR ALL ALGORITHMS

Algorithms	***nb***	**Results of Decryption Time in *Sec.* for all algorithms at all cases of *d1* and *nb***			
		d1=4	***d1=8***	***d1=16***	***d1=32***
Arnold-based Algorithm	***nb=1***	5.3200	1.3570	0.3590	0.1090
	nb=2	2.6990	0.7020	0.2030	0.0690
VSA3DCS with Lorenz	***nb=1***	0.1090	0.0780	0.0780	0.0870
	nb=2	0.0820	0.0780	0.0780	0.0680
VSA3DCS with Chen's	***nb=1***	0.0860	0.0830	0.0830	0.0780
	nb=2	0.0780	0.0860	0.0780	0.0680

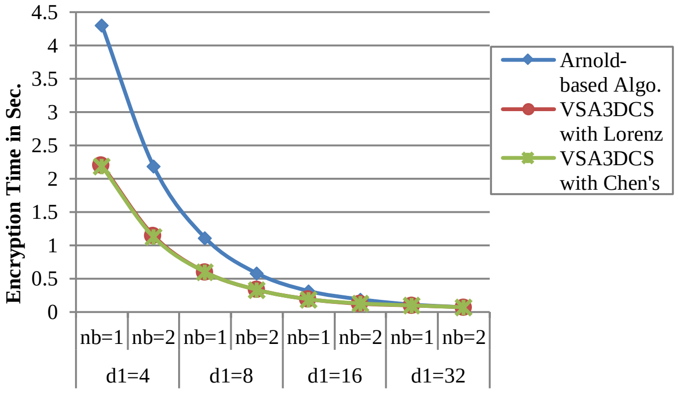


Fig 18. Encryption Time in Sec. of applying all Algorithms at all Cases.

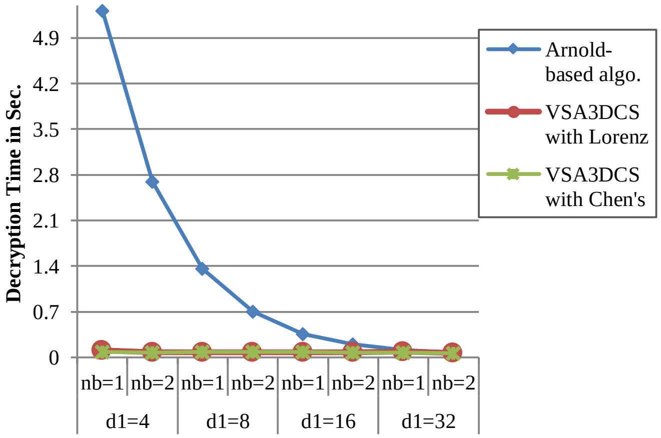


Fig 19. Decryption Time in Sec. of applying all Algorithms at all Cases.

Tables I and II, with Fig. 18 and 19, illustrate the execution time of encryption and decryption of the VSA3DCS algorithm with both Lorenz and Chen's is less than the time encryption and decryption of the Arnold-based algorithm at all cases of *d1* and *nb*. So, the VSA3DCS algorithm with both Lorenz and Chen's is better than the Arnold-based algorithm in all cases of *d1* and *nb*.

B. Correlation Coefficient Measure

If encrypted and original files are highly correlated, the coefficient of correlation equals one, i.e. the encryption method is ineffective in hiding the original signal information. If the coefficient of correlation is equal to zero then the initial voice signals and its encryption are entirely different. Progress of the encryption method thus implies lower CC values. The CC is computed using formula (12) [1,2,6,7]:

$$CC = \frac{\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y))}{\sqrt{\sum_{i=1}^{N}(x_i - E(x))^2}\sqrt{\sum_{i=1}^{N}(y_i - E(y))^2}} \qquad (12)$$

Table III shows the results of CC analysis for encrypting by applying all algorithms; Arnold-based algorithm, and VSA3DCS with both Lorenz and Chen's on original audio signals at all cases of *d1* and *nb*. Fig. 20 illustrates the plot for the results of CC for scrambled audio signals produced by all algorithms at all cases of *d1* and *nb*.

TABLE III. RESULTS OF CC ANALYSIS FOR ENCRYPTING WITH ALL ALGORITHMS IN ALL CASES

Algorithms	***nb***	**Results CC for all Algorithms at all cases of *d1* and *nb***			
		d1=4	***d1=8***	***d1=16***	***d1=32***
Arnold-based Algorithm	***nb=1***	0.0046	-0.0116	-0.0086	-0.0051
	nb=2	0.0045	-0.0117	-0.0085	-0.0050
VSA3DCS with Lorenz	***nb=1***	-0.0026	0.0042	0.0031	0.0011
	nb=2	-0.0018	-0.00086	0.000406	-0.0035
VSA3DCS with Chen's	***nb=1***	0.000081	-0.0019	0.0033	-0.0034
	nb=2	0.0026	-0.0012	0.0021	0.0050

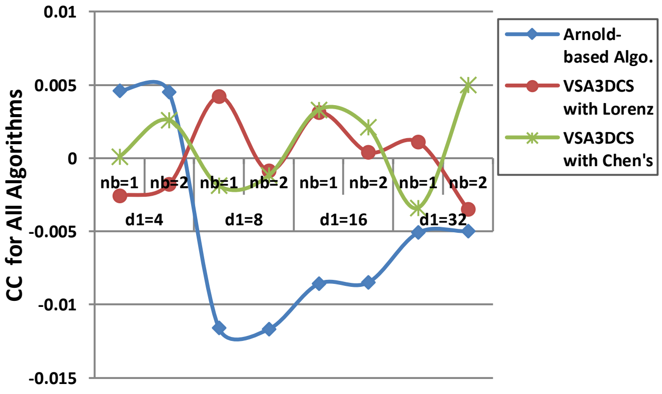


Fig 20. The CC of applying all Algorithms in all Cases.

Table III and Fig. 20 illustrate that the VSA3DCS algorithm with both Lorenz and Chen's achieves very small values (near to zero) of CC compared with the results of the Arnold-based algorithm, i.e., the VSA3DCS algorithm with both Lorenz and Chen's better than the Arnold-based algorithm. So, the proposed algorithm VSA3DCS is complex and strong for the encryption of the audio signal.

Results of the CC of decrypted audio signals equal to 1 for all algorithms in all cases of *d1* and *nb*, because decryption by all algorithms returns the decrypted audio signals file completely matched to the original audio signals file.

C. Spectral Distortion (SD) Measure

The SD is a type of measurements implemented in the frequency spectra of original and encrypted audio signals within the frequency domain. In dB it is calculated to demonstrate how far from that of the original audio signals the encrypted signal range is. The SD is calculable as in formula (13) [5,18]:

$$SD = \frac{1}{M}\sum_{m=0}^{M-1}\sum_{n=L,m}^{L,m+L,-1}\left| V_s(k) - V_y(k)\right| \qquad (13)$$

Where $V_s(k)$ is the spectrum of the primary audio signal in dB for a given portion, $V_y(k)$ is the spectrum of the encoded/decoded audio signal in dB for the same portion, *M* is the number of portions and *L* is the duration of the portion. The bigger the SD between the original and encrypted signals, the greater the encryption efficiency. On the other hand, between the primary audio signals and the decrypted signals, The SD must be as small as possible.

Table IV shows the results of the SD measure for encrypting by applying all algorithms in all cases of *d1* and *nb*. And, Fig. 21 displays the results of SD for encrypted audio signals produced by applying all algorithms at all values of *d1* and *nb*.

Table IV and Fig. 21 illustrate that all algorithms (VSA3DCS with both chaotic systems and Arnold) achieve good values for SD at all cases of *d1* and *nb*, whereas all results bigger than *13.91* (far from zero), so all of them are complex and strong algorithms for audio signals encryption. But in the most cases, the results of VSA3DCS with both chaotic systems is greater and better than the results of the Arnold-based algorithm.

TABLE IV. RESULTS OF SD MEASURE FOR ENCRYPTING WITH ALL ALGORITHMS AT ALL CASES

Algorithms	***nb***	**Results SD for all algorithms at all cases of *d1* and *nb***			
		d1=4	***d1=8***	***d1=16***	***d1=32***
Arnold-based Algorithm	***nb=1***	14.0178	13.9143	13.9700	13.9587
	nb=2	14.0046	13.9322	13.9867	13.9371
VSA3DCS with Lorenz	***nb=1***	14.0000	14.0400	14.0100	14.0000
	nb=2	14.0000	14.0376	13.9886	13.9500
VSA3DCS with Chen's	***nb=1***	14.0100	14.0220	13.9800	14.0400
	nb=2	14.0223	13.9700	14.1000	14.1176

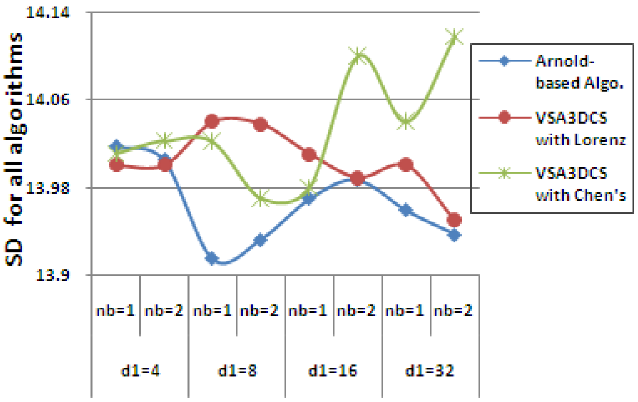


Fig 21. The SD of applying all Algorithms at all Cases.

Results of SD for decrypted audio signals equal to *0* with all algorithms at all cases, because decrypted audio signals file completely matched to the original audio signals file.

D. Log-Likelihood Ratio (LLR) Measure

The Audio signal LLR metric is based on the assumption that each component can be interpreted through a predictive linear all-pole model of the formula (14) [5,18]:

$$S(n) = \sum_{m=1}^{m_p} a_m s(n-m) + G_s u(n) \tag{14}$$

where a_m (for m=1, 2, ….., m_p) are all-polar filter coefficients, G_s is the filter gain and $u(n)$ is a good source of excitation for the filter. The audio signal is fenced to form frames have lengths of 15 to 30ms. LLR metric is then determined as in [5]:

$$LLR = \left|\log\left(\frac{\vec{a}_s \vec{R}_y \vec{a}_s^T}{\vec{a}_y \vec{R}_y \vec{a}_y^T}\right)\right| \tag{15}$$

where, $\vec{\mathbf{a}_s}$ is the coefficient vector for LPCs; [*1*, $a_s(1)$, $a_s(2)$, . ., a_s (m_p)] for the premier clear audio signal, $\vec{\mathbf{a}_y}$ is the coefficient vector for LPCs; [1, $a_y(1)$, $a_y(2)$, …… , $a_y(m_p)$] for the encryption/decrypted audio signals, and $\vec{\mathbf{R}_y}$ is the autocorrelation matrix of the encryption/decrypted audio signals. The higher the LLR between the original and the encrypted signals, the greater the encryption efficiency. In comparison, the lower the LLR is to zero, the greater the decryption efficiency.

TABLE V. RESULTS OF LLR MEASURE FOR ENCRYPTING WITH ALL ALGORITHMS IN ALL CASES

Algorithms	***nb***	**Results LLR for all algorithms in all cases of *d1* and *nb***			
		d1=4	***d1=8***	***d1=16***	***d1=32***
Arnold-based Algorithm	***nb=1***	0.7028	0.5524	0.6776	0.5927
	nb=2	0.4741	0.4674	0.4272	0.3375
VSA3DCS with Lorenz	***nb=1***	0.7746	0.7763	0.7671	0.8371
	nb=2	0.7133	0.7207	0.8081	0.6738
VSA3DCS with Chen's	***nb=1***	0.7238	0.7000	0.7799	0.7694
	nb=2	0.7463	0.7133	0.7415	0.7144

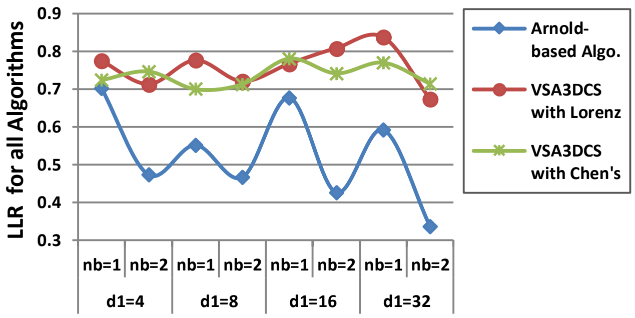


Fig 22. The LLR of applying all Algorithms in all Cases.

Table V shows the results of LLR measure for encrypting by applying all algorithms in all cases of *d1* and *nb.* Fig. 22 displays results of LLR of encrypted audio signals generated by all algorithms in all cases.

Table V and Fig. 22 illustrate that the VSA3DCS algorithm with both Lorenz and Chen's achieves very good results for LLR in all cases of *d1* and *nb*, i.e., the LLR results with VSA3DCS algorithm are better than the results with Arnold-based algorithm at all cases of *d1* and *nb.* So, the proposed algorithm VSA3DCS are complex and strong algorithm for audio signal encryption.

Results of LLR of decrypted audio signals equal to *0* for all algorithms at all cases of *d1* and *nb*, because the decrypted audio signals file completely matched to the original audio signals file.

E. Key Sensitivity Measure

The experimental results indicate that both the Arnold-based algorithm and the VSA3DCS algorithm with both Lorenz and Chen's are extremely sensitive to the mismatching of hidden keys. Table VI displays keys sensitivity results for all algorithms.

From Table VI, we can see that the VSA3DCS algorithm with both Lorenz and Chen's has greater space for the keys than the Arnold-based algorithm. Also, any of the keys with little movement (e.g., 10^{-17} is modified to *h*) will generate an incorrect decrypted image. VSA3DCS algorithm is therefore very sensitive to the keys, and they can also withstand various sensitivity dependent attacks.

TABLE VI. RESULTS OF KEY SENSITIVITY MEASURING FACTOR FOR ALL ALGORITHMS

*Name and precision For **Arnold-based Algorithm***							
q				***p***			
10^{-15}				10^{-15}			
*Name and precision For **VSA3DCS with Chen's***							
a	***b***	***c***	***h***	***k***	***x_0***	***y_0***	***z_0***
10^{-14}	10^{-15}	10^{-14}	10^{-17}	10^{-10}	10^{-16}	10^{-15}	10^{-14}
*Name and precision For **VSA3DCS with Lorenz***							
σ	***r***	***b***	***h***	***k***	***x_0***	***y_0***	***z_0***
10^{-15}	10^{-14}	10^{-15}	10^{-17}	10^{-10}	10^{-16}	10^{-15}	10^{-14}

Table VI illustrates the results of the precision of the keys for the VSA3DCS algorithm are better than the results of the Arnold-based algorithm. Therefore, VSA3DCS satisfies high quality of security better than the other.

VII. Conclusion

In this paper, a proposed voice scrambling algorithm (VSA3DCS) based on one of 3D chaotic maps systems (Lorenz or Chen's) is presented and compared with the Arnold-based algorithm. VSA3DCS algorithm and Arnold chaotic algorithm are applied on audio signals file to encrypt it by scrambling process for its signals' positions. The encrypted audio signals which produced from applying all algorithms are compared and discussed by using some experiential measures and comparative analysis, such as; the encryption/decryption time, the Correlation Coefficient (CC) of the evident and encrypted signals between samples, the Spectral Distortion (SD) measure, Log-Likelihood Ratio (LLR) measure, and key sensitivity measure. The encryption/decryption time for all algorithms is very good, but the VSA3DCS algorithm with both Lorenz and Chen's achieves encryption/decryption time very close to zero and less and better than encryption/decryption time of Arnold-based algorithm in all cases of *d1* and *nb*. The results of CC are better with the VSA3DCS algorithm than the other with the Arnold-based algorithm in all cases of *d1* and *nb*. In the results of SD, in the most cases, the results of VSA3DCS with both chaotic systems are greater and better than the results of the Arnold-based algorithm. The results of LLR are better with the VSA3DCS algorithm than the other with the Arnold-based algorithm at all cases of *d1* and *nb*. Also, the VSA3DCS algorithm with both Lorenz and Chen's has greater space for the keys than the Arnold-based algorithm, also, the VSA3DCS algorithm is very sensitive to the keys. Also, the plots of scrambled audio signals' patterns and spectrogram illustrate the VSA3DCS algorithm with both Lorenz and Chen's is better than the Arnold-based algorithm. The final results show that the VSA3DCS algorithm is a strong algorithm to supply an efficient and stable approach for encrypting audio signals.

References

[1] Osama M. Abu Zaid, Nawal F El Fishawy, Elsayed Nigm “ Encryption Quality Measurement of a Proposed Cryptosystem Algorithm for the Colored Images Compared with Another Algorithm,” Arab J. Inf. Technol. 13 (1), pp. 20-29, Jan. 2016.

[2] El-Fishawy N., Osama M. Abu Zaid, "Quality of Encryption Measurement of Bitmap Images with RC6, MRC6, and Rijndael Block Cipher Algorithms," International Journal of Network Security, vol. 5, no. 3, pp. 241–51, 2007.

[3] Roayat Ismail Abdelfatah, Mohamed E. Nasr, and Mohammed A. Alsharqawy, " Encryption for Multimedia Based on Chaotic Map: Several Scenarios," Multimedia Tools and Applications, Springer Nature, https://doi.org/10.1007/s11042-020-08788-8, 2020.

[4] Osama M. Abu Zaid, El-fishawy N., Nigm E., Faragallah O., "A Proposed Encryption Scheme Based on Henon Chaotic System (PESH) for Image Security," International Journal of Computer Applications, vol. 61, no. 5, pp.29-39, 2013.

[5] E. Mosa, N.W. Messiha, O. Zahran, F.E. Abd El-Samie. Dec.2011. "Chaotic encryption of speech signals", International Journal of Speech Technology,14(4), pp. 285-296, 2011.

[6] Ankita Bisht, Mohit Dua, Shelza Dua and Priyanka Jaroli, "A Color Image Encryption Technique Based on Bit-Level Permutation and Alternate Logistic Maps," Journal of Intelligent Systems , vol. 29, no. 1, pp.1246–1260, 2020.

[7] Chao-Feng Zhao, and Hai-Peng Ren, " Image Encryption Based on Hyper-Chaotic Multi-Attractors," Nonlinear Dynamics, Springer, vol. 100, pp.679–698, 2020.

[8] Mahmoud F. Abd Elzaher, Mohamed Shalaby, and Salwa H. El Ramly, "An Arnold Cat Map-Based Chaotic Approach for Securing Voice Communication," INFOS '16: Proceedings of the 10th International Conference on Informatics and Systems, pp. 329–331, May 2016.

[9] Chong Fu, Wen-jing Li, Zhao-yu Meng, Tao Wang, and Pei-xuan Li, " A Symmetric Image Encryption Scheme Using Chaotic Baker map and Lorenz System," Ninth International Conference on Computational Intelligence and Security 2013, IEEE Xplore, pp. 724-728, 2013.

[10] Huibin Lu, Xia Xiao., "A Novel Color Image Encryption Algorithm Based on Chaotic Maps," Advances in Information Sciences and Service Sciences (AISS), Vol. 3, No. 11, pp. 28-35, December 2011.

[11] Osama M. Abu Zaid, Nawal F El Fishawy, Elsayed Nigm “Cryptosystem Algorithm Based on Chaotic System for Encrypting Colored Image,” International Journal of Computer Science Issues 10 (4), pp. 215-224, 2013.

[12] M Demba, and Osama M. Abu Zaid, “A Proposed Confusion Algorithm Based on Chen’s Chaotic System For Securing Colored Images,” International Journal of Signal Processing Systems Vol. 1, No. 2, pp. 296-301, December 2013.

[13] Osama M. Abu Zaid, Nawal F El Fishawy, Elsayed Nigm “A Proposed Permutation Scheme Based on 3-D Chaotic System,” International Journal of Computer Science Issues 10 (4), pp. 208-214, 2013.

[14] Zhenwei Shang, Honge Ren, Jian Zhang., " A Block Location Scrambling Algorithm of Digital Image Based on Arnold Transformation ", the 9th International Conference for Young Computer Scientists (ICYCS2008), IEEE Xplore, pp. 2942-2947, 2008.

[15] Anjana Savita, Phool Singh, A K Yadav, and Kehar Singh, "Asymmetric Audio Encryption System Based on Arnold Transform and Random Decomposition", Asian Journal of Physics, Vol. 27, Nos 9-12, pp. 711-719, 2018.

[16] Zhi-Hong Guan, Fangjun Huang, and Wenjie Guan, "Chaos-Based Image Encryption Algorithm," Physics Letters A, Vol. 346, pp. 153-157, 2005.

[17] Yuanzhi Wang, Guangyong Ren, Julang Jiang, Jian Zhang, and Lijuan Sun, "Image Encryption Method Based on Chaotic Map," 2007 Second IEEE Conference on Industrial Electronics and Applications, IEEE Xplore, pp. 2558-2560, 2007.

[18] E. Mosa, N.W. Messiha, O. Zahran, F.E. Abd El-Samie, "Encryption of Speech Signal with Multiple Secret Keys in Time and Transform Domains", International Journal of Speech Technology, Springer, Vol. 13, No. 4, pp. 231–242, 2010.