An audio encryption technique through compressive sensing and Arnold transform

Nishanth Augustine*

Department of ECE,
LBS College of Engineering,
Kasaragod, India
Email: nishanthaugustine@gmail.com
*Corresponding author

Sudhish N. George and Deepthi P. Pattathil

Department of ECE,
National Institute of Technology,
Calicut, India
Email: sudhish@nitc.ac.in
Email: deepthi@nitc.ac.in

Abstract: In this paper, an audio encryption scheme using compressive sensing (CS) and Arnold transform-based scrambling is presented. In the proposed method, compressive sensing is done by using a key-based measurement matrix and the scrambling is carried out with the help of a key-based Arnold matrix. The use of these key-based matrices not only provides better security but also do away with their transmission and storage requirement. The measurement matrix is constructed by using the random numbers generated by a linear feedback shift register (LFSR) whose initial state is generated by a piece wise linear chaotic map (PWLCM), using three 32-bit keys whereas the Arnold matrix is constructed by the random numbers generated by using another 32-bit secret key, PWLCM and a logistic map. By combining secure compressive sensing and Arnold scrambling techniques, very high security can be ensured in addition to efficient channel usage, good resistivity to noise, best reconstruction performance, little encoder complexity and excellent scrambling of data. Experimental results prove the effectiveness of the proposed scheme.

Keywords: encryption; compressive sensing; Arnold transform; scrambling degree; residual intelligibility; key-space; brute force attack; known plain-text attack; perceptual evaluation quality; robustness.

Reference to this paper should be made as follows: Augustine, N., George, S.N. and Pattathil, D.P. (2015) 'An audio encryption technique through compressive sensing and Arnold transform', *Int. J. Trust Management in Computing and Communications*, Vol. 3, No. 1, pp.74–92.

Biographical notes: Nishanth Augustine graduated in Electronics and Communication Engineering from Cochin University of Science and Technology, Kerala in 2004, and MTech in Signal Processing from the National Institute of Technology Calicut in 2014. He is working as an Assistant Professor at LBS College of Engineering, Kasaragod, India. His interests include audio processing and multimedia security.

Sudhish N. George received his BTech in Electronics and Communication Engineering from M.G University, Kerala, India, in 2004, MTech in Signal Processing from Kerala University, India, in 2007, and PhD degree from the National Institute of Technology Calicut, India in the field of multimedia security in 2014. He is working as an Assistant Professor in the Department of Electronics and Communication, National Institute of Technology Calicut from 2010 onwards. His current interests include signal processing and multimedia security.

Deepthi P. Pattathil received her BTech in Electronics and Communication Engineering from N.S.S.College of Engg, Palakkad (Calicut University) in 1991, MTech in Instrumentation from the Indian Institute of Science, Bangalore in 1997, and PhD degree from the National Institute of Technology Calicut in the field of secure communication in 2009. She has been working as faculty in institutions under IHRD, Thiruvanthapuram from 1992 to 2001 and in the Department of Electronics and Communication Engineering, National Institute of Technology Calicut from 2001 onwards. Her current interests include cryptography, signal processing with security applications, information theory and coding theory.

This paper is a revised and expanded version of a paper entitled 'Compressive sensing based audio scrambling using Arnold transform' presented at Second international Conference on Security in Computer Networks and distributed Systems (SNDS'14), Trivandrum, 13–14 March 2014.

1 Introduction

A communication scheme is said to be secure if it is capable to protect the data from any kind of eavesdropping. Encryption is an effective means of providing information security against illegal surveillance and wire tapping. One of the main technologies that has been developed to obscure the content of transmission is the time domain scrambling. In the time domain scrambling process, a segment of time domain samples are taken and scrambled into a different segment of samples. This scrambled data is transmitted and it is descrambled at the receiving end to its original form. The scrambling and descrambling operations are based on a scrambling/descrambling matrix (Zeng et al., 2012).

Earlier audio scrambling matrices were constructed by pseudo-random sequences (Lin and Abdulla, 2007), Hadamard transform (Senk et al., 1997), Fibonacci transform (Nan et al., 2004) and so on. The disadvantage of these matrices is that since the matrix is invariable, these methods could easily be deciphered. Some improved algorithms such as stochastic matrix (Li et al., 2010) and Latin square (Satti and Kak, 2009) were developed to overcome this problem, but they result in heavy transmission load. Speech compression methods like G.729 mixed excitation linear prediction (MELP) and adaptive multi-rate (AMR) (Servetti and De Martin, 2002) audio codec are then employed along with the process of scrambling to reduce the transmission load, but these methods shows low robustness in the presence of noise. Thus, the disadvantages of earlier audio scrambling matrices are their easiness in decipherability, heavy transmission load and low robustness. Key-based scrambling matrix construction reduces the transmission load and makes the data unable to decipher easily.

One of the major techniques used for changing the location of a point randomly is the Arnold transformation technique (Shang et al., 2008). An algorithm for image scrambling, using two dimensional Arnold transform was proposed in Huang et al. (2012) which uses a 64-bit key and a logistic map for constructing the Arnold matrix which is then used to rearrange the pixels. The same algorithm can also be used to scramble the audio data. Arnold scrambling offers excellent scrambling degree (SD) (Madain et al., 2012) and breaks the correlation between audio samples effectively.

In wireless multimedia sensor networks (WMSN), the encoder complexity and the bandwidth requirement should be minimised. The emerging technology, compressive sensing (CS) (Donoho, 2006) focuses in this direction. CS performs both sampling as well as compression of the source information simultaneously. It represents a signal using a number of linear, non-adaptive measurements which is much lower than the number of samples needed if the signal is sampled at the Nyquist rate provided the signal is sparse in some basis. CS stores and transmits only a few non-zero coefficients and enables the recovery of signals from these minimum measurements. This significantly reduces the time of data acquisition, storage requirement and the amount of data needed to be transmitted (Donoho, 2006). The random measurements are taken by using a measurement matrix of suitable size and the receiver should know this matrix for the reconstruction. In key-based CS, the measurement matrix is constructed according to a secret key and this makes the data undecipherable (Orsdemir et al., 2008). In addition to this, CS provides high robustness against noise and very good data compression. Key-based measurement matrix construction eliminates the necessity of transmission and storage of the same.

The residual intelligibility and key space are the measures that are used to evaluate the degree of security of a scrambling algorithm (Del Re et al., 1989). Residual intelligibility is the amount of intelligibility left over in the scrambled signal. The lower the residual intelligibility of a scrambling method, the higher its degree of security. SD can be used to evaluate the degree of security. As SD increases, degree of security increases and residual intelligibility decreases. Key space is the number of keys available for scrambling. Larger key space provides higher degree of security. An efficient scrambling method should be channel-saving, attack-resistant and should provide high SD. Since Arnold scrambling (Shang et al., 2008) provides very good SD whereas CS provides very good compression and robustness, by combining both these techniques, a secure audio communication scheme can be developed.

The audio scrambling scheme proposed in our previous work (Augustine et al., 2014) combines CS and Arnold scrambling. It employs key-based measurement matrix generation proposed in Orsdemir et al. (2008) for CS and key-based Arnold matrix generation proposed in Huang et al. (2012) for scrambling. This algorithm is modified by incorporating a piece wise linear chaotic map (PWLCM), linear feedback shift register (LFSR) and a logistic map for improving security.

In the proposed method, first CS is applied on the audio signal using a measurement matrix created by using secret keys, PWLCM and LFSR and then the resultant lower dimensional vector is scrambled using Arnold matrix constructed with the help of a secret key, PWLCM and a logistic map. By performing CS first, the amount of data to be scrambled can be reduced significantly and this results in considerable reduction in computations.

The rest of this paper is organised as follows. Section 2 illustrates the basics of Arnold transform and CS. The proposed scheme is explained in Section 3. Security

analysis is presented in Section 4. Section 5 gives the analysis and discussion of the experimental results. The proposed method is compared with other schemes in Section 6 and conclusions are drawn in Section 7.

2 Fundamentals of Arnold transform and CS

2.1 Arnold transform

Arnold transform is a transformation technique which is commonly used in image scrambling to rearrange the pixels of the image randomly. It is actually a location moving of a point. In the case of two-dimensional Arnold transform (Shang et al., 2008), if (x, y) is a point in a matrix of size $p \times q$, then the transformation that change the point (x, y) to another point (x', y') is given by

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \bmod \begin{bmatrix} p \\ q \end{bmatrix}$$

This is an equal area transform and it can be iterated. Arnold transform is cyclical, that is, when iterate to a certain step, it will regain the original location. Since there are many methods for calculating the periodicity and getting the inverse transform, the use of traditional Arnold transform for the scrambling has become unsafe. So the traditional Arnold transform is modified by adding two parameters a and b, where a and b are positive integers and the new transform is

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \left(\begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \right) \bmod \begin{bmatrix} p \\ q \end{bmatrix} \tag{1}$$

We can choose different transform coefficient a and b and it is difficult to regain the original location after the transform because the transform coefficient is not the only. This improves the efficiency of scrambling algorithm and security of the data (Shang et al., 2008).

2.2 Compressive sensing

CS (Shang, 2006), relies on the sparsity of the signal. The basic idea behind CS is that the signals that are composed as the linear combinations of few linearly independent vectors need only to be sampled at a low rate to facilitate a high quality reconstruction (Shang, 2006). Here, few means that the number of basis vectors is small relative to the number of samples required if it is sampled at Nyquist rate. By employing CS, a great majority of the data can be compressed. CS theory is based on the assumption that the signal of interest is sparse in some basis as it can be accurately and efficiently represented in that basis.

CS uses random measurements in a basis that is incoherent with the sparse basis. Incoherence (Tropp, 2004) means that no element of one basis has a sparse representation in terms of the other basis. CS has found applications in many areas such as image processing, spatial localisation, medical signal processing, etc.

Consider a signal X which is an $N \times 1$ vector. Let X be sparse in some another domain Ψ such that

$$X = \Psi v \tag{2}$$

where v is the sparse coefficients of X which is an $N \times 1$ vector, but with only K coefficients ($K << N$) are non-zero. Ψ is called dictionary matrix and of size $N \times N$. i.e., X can be represented by using only K coefficients in Ψ.

The main idea of CS is to map the observed signal X to a lower-dimensional vector Y via measurement matrix Φ by the following transformation:

$$Y = \Phi X \tag{3}$$

where Φ is an $M \times N$ matrix ($K < M < N$) and should satisfy restricted isometric property (RIP) (Candès and Wakin, 2008) and should be incoherent with Ψ.

To do reconstruction, we need to introduce a sparsity metric on v. A commonly used measure for this is the l_1-norm (Candès and Wakin, 2008) denoted as $\|.\|_1$, which can be related to the number of non-zero coefficients under certain technical conditions. The vector v is said to be K sparse if it contains only K non-zero coefficients. We can now pose the sparse decomposition problem as the following:

$$\text{minimise} \| v \|_1 \text{ s.t. } X = \Psi v \tag{4}$$

Similarly, the measurement matrix Φ is also multiplied on to Ψv, and we can write the sparse decomposition problem as

$$\text{minimise} \| v \|_1 \text{ s.t. } Y = \Phi \Psi v \tag{5}$$

introducing $\Theta = \Phi\Psi$, above problem can be rewritten as

$$\text{minimise} \| v \|_1 \text{ s.t. } Y = \Theta v \tag{6}$$

The key point of CS theory is that under certain conditions, namely an appropriate choice of measurement matrix Φ, solving (2) will result in a solution vector v' identical to that of above equation. Therefore, reconstruction of the signal using v' obtained will reconstruct not only Y, but also X exactly as $X = Yv'$ when v is sparse.

3 Proposed method

The proposed scheme has to perform two processes: CS and scrambling. First, the audio signal is compressed and encrypted by taking random measurements of original samples using a key-based measurement matrix, and then the compressed vector is scrambled using Arnold transform. A block diagram representation of the encoder section is shown in Figure 1.

The scrambling algorithm takes the compressively sensed audio file as the input and produces a scrambled output. The compressed and scrambled file is transmitted and at the receiver side this file is descrambled and the original file is reconstructed from this descrambled data. Two methods are used for reconstruction: one convex optimisation algorithm, l_1-minimisation (Tropp, 2004) and one greedy approach, the orthogonal matching pursuit (OMP) (Tropp, 2004).

The degree of security of the proposed method relies on four keys; three 32 bit keys for the measurement matrix generation and a 32 bit key for the Arnold matrix generation. Thus, the key space is 2^{128}, which is enough for providing very high security in the case of multimedia applications.

3.1 key-based measurement matrix generation

The random measurements of the original samples are taken by multiplying them by a measurement matrix of suitable size and the receiver should know this matrix to reconstruct the original signal. If the matrix construction is based on a key, the data becomes secure and the transmission requirement of the matrix can be eliminated. A measurement matrix construction procedure using LFSR is proposed in George and Pattathil (2014). According to this, LFSR is loaded with an initial state and in each clock, the state of LFSR is modified according to a feedback polynomial (Krawczyk, 1994). The states of LFSR are converted to their decimal equivalent values and are used as the entries of a matrix of order $M \times N$. Orthonormalisation of this matrix will results in a matrix having the desired features of measurement matrix such as RIP and incoherence and hence, it can be used for taking random measurements of the original signal. To improve the security, it is proposed to generate the initial state of LFSR using secret keys through PWLCM.

3.2 Initial state generation of LFSR

In the proposed scheme, the LFSR is initially loaded with the binary sequence constructed from a random number, generated by running PWLCM for a number of iterations. The system parameter, initial value and the number of iterations of PWLCM are formed by using three 32-bit keys, key_1, key_2 and key_3 and are kept as secret. These keys are chosen in such way that they do not divide each other. i.e.,

$$Key_i \nmid Key_j, \quad \text{for } i \neq j \quad i, j \in \{1, 2, 3\} \tag{7}$$

PWLCM is a chaotic sequence model which is more chaotic than normal logistic map. The mapping function of a one dimensional PWLCM for the $(n + 1)^{th}$ iteration is given by

$$x_{n+1} = F(x_n) = \begin{cases} x_n / \gamma, & 0 \le x_n < \gamma \\ (x_n - \gamma)/(0.5 - \gamma), & \gamma \le x_n < 0.5 \\ F(1 - x_n), & 0.5 \le x_n < 1 \end{cases} \tag{8}$$

where $_{xn}$ is the state of PWLCM for the n^{th} iteration and γ is the system parameter. The value of γ is bounded in the interval (0, 0.5) and x_n is chaotic and ergodic in the interval [0, 1). The initial value x_0, system parameter γ and the number of iterations Q of PWLCM are calculated using the keys as given below:

$$x_0 = \left(\frac{Key_{1d}}{Key_{2d}} \right) - \left\lfloor \frac{Key_{1d}}{Key_{2d}} \right\rfloor \tag{9}$$

$$\gamma = \frac{\left(\frac{Key_{2d}}{Key_{3d}}\right) - \left\lfloor \frac{Key_{2d}}{Key_{3d}} \right\rfloor}{2} \tag{10}$$

$$Q = \left(5key_{3d} + 1\right) \bmod 128 \tag{11}$$

where key_{1d}, key_{2d} and key_{3d} are the decimal equivalents of key_1, key_2 and key_3 respectively. Running PWLCM, Q number of times using initial value x_0 and parameter γ results in a random number x_Q, whose value is within the interval [0, 1).

To generate binary sequence from this number, first fix a threshold value T. Let L be the length of LFSR. Multiply x_Q by two and check whether it is below the threshold. If so, the binary bit is 0 otherwise it is 1. Repeating this L times results in L bits. These bits are then used to initialise the LFSR.

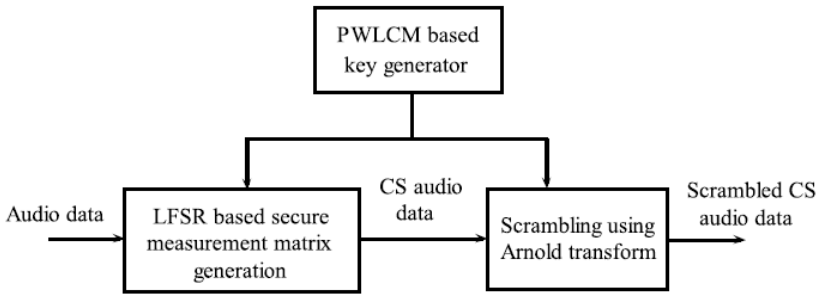
3.3 Arnold matrix generation

The Arnold matrix, used for scrambling is constructed by using a 32-bit key (key_4) and a logistic map. First step is to create a 64-bit binary stream from key_4 through PWLCM. The same algorithm described in the above section can be used for this sequence creation. The only difference is that here the number of iterations Q' is calculated using key_4 using the equation

$$Q' = \left(5key_{4d} + 1\right) \bmod 128 \tag{12}$$

and making $L = 64$. (key_{4d} is the decimal equivalents of key_4).

Figure 1 Encoder section



The logistic map is a polynomial mapping which can be expressed mathematically as $x_{n+1} = \mu x_n[1 - x_n]$, where x_n is a number between zero and one and μ is a positive number. A logistic map exhibit a great sensitivity to initial conditions if the values of μ is in between about 3.57 and 4 and hence, it can be considered as a chaotic system. The logistic map scheme is reliable, unpredictable, offers randomness and does not require any computationally intensive algorithms. The steps for Arnold matrix generation are given below (Huang et al., 2012):

1 Create 64-bit binary stream using key_4 through PWLCM.

2 Divide this bit stream into two 32 bit binary numbers and then find their decimal equivalents, K_1 and K_2 and their sum K_T.

3 Calculate $I_1 = K_1/K_T$ and $I_2 = K_2/K_T$.

4 Iterate I_1 and I_2, T times using the logistic map $I_j(i) = \mu I_j(i)[1 - I_j(i)]$ for $j = 1, 2$ and $i = 1, 2, \ldots, T$.

5 Identify the most significant three bits, b_{1j}, b_{2j}, b_{3j} of I_j for $j = 1, 2$.

6 Generate two decimal numbers a_1 and a_2 using the equation $a_j = 100_{b1j} + 10_{b2j} + b_{3j}$ for $j = 1, 2$ and calculate their product a_3.

7 Construct Arnold matrix, A using the above values as

$$A = \begin{bmatrix} 1 & a_1 \\ a_2 & a_3 + 1 \end{bmatrix}$$

3.4 Proposed encryption algorithm

The procedure starts by reading an audio file and determining its length, N. Choose suitable sub-rate, S (sub-rate is the ratio between the length of compressed file to that of original file) and calculate the number of random measurements required, M ($M = N \times S$). The size of the transformed file is reduced to M by taking random measurements, i.e., by multiplying with a measurement matrix of size $M \times N$ which is composed of numbers, generated randomly by an LFSR which is initialised by using three 32-bit keys. Once the signal has gone through the process of CS then scramble it using Arnold transformation. For that reshape, the 1D measurement vector to a 2D array having M cells. Construct the 2D Arnold matrix by the numbers generated by using the 64-bit key and logistic map. The new indices values used for scrambling is obtained by multiplying the current indices values with the Arnold matrix, K times. The compressed audio samples are transformed into another two dimensional array according to this indices list. After filling all audio samples, the two dimensional matrix is converted into one dimensional array. This scrambled audio file is written with the same sample rate and number of bits per sample as its original. The algorithm can be described as follows:

1 Read audio file X and determine its length N.

2 Fix a sub-rate S and calculate M.

3 Using the three 32-bit keys, calculate the initial value x_0, system parameter γ and number of iterations Q of PWLCM.

4 After running PWLCM Q times, generate the binary sequence using x_Q.

5 Initialise LFSR using this binary sequence and modify its states $M \times N$ times.

6 Compute the decimal equivalent of each and every states of LFSR and construct the measurement matrix Φ of size $M \times N$ using these numbers.

7 Take M random measurements of X' by multiplying it with the measurement matrix Φ.

8 Reshape the measurement vector Y into a rectangular matrix $Y1$ of size $p \times q$, where $p \times q = M$.

9 Construct Arnold matrix A using 32-bit key, PWLCM and logistic map.

10 Calculate the new index values, (x', y') by multiplying the current index values, (x, y) with Arnold matrix using equation (1) for $x = 1, \ldots, p$ and $y = 1, \ldots, q$.

11 Repeat above step R times.

12 Construct another matrix $Y2$ such that $Y2(x', y') = Y1(x, y)$.

13 Reshape $Y2$ to a 1D sequence Z of size $M \times 1$.

14 Write the scrambled file Z in the same format, with the same sample rate and number of bits per sample as that of X.

3.5 Decryption algorithm

It takes the scrambled file Z, descrambles it and original audio file is reconstructed from this descrambled file. The descrambling process is similar to that of scrambling process. The only difference is that the descrambling matrix is the inverse of Arnold matrix, A. The audio file, X_{rec} can be reconstructed by applying l_1-minimisation or OMP to the descrambled file.

4.6 Computational complexity

The major tasks to be performed in this audio encryption scheme are the construction of measurement matrix and Arnold matrix, CS and scrambling and descrambling of compressive sensed data. The matrices are constructed in advance and the operations to be performed at the transmitter side in real-time are the CS and scrambling. Let N be the length of audio file, S be the sub-rate and M be the length of compressive sensed file. The number of arithmetic operations required to perform in real-time are two (one $[M \times N] * [N \times 1]$ and one $[2 \times 2] * [2 \times 1]$) matrix multiplications. For example, let $N = 1{,}000$, $S = 0.1$ and $R = 10$ and hence $M = 100$. The required operations are one $[100 \times 1{,}000] * [1{,}000 \times 1]$ and one $[2 \times 2] * [2 \times 1]$) matrix multiplications. These operations will take only a few milliseconds and can be done in real-time. By using a greedy algorithm at the receiver side, the reconstruction can also be performed in real-time.

4 Security analysis

In this section, the security of the proposed encryption method against different attacks like brute force attack and known plain text attack are analysed.

4.1 Brute force attack

In this encryption method, the degree of security relies on four 32-bit keys; three are used for constructing the measurement matrix of CS and the remaining one for constructing the Arnold matrix used for scrambling. The initial value x_0 and system parameter γ of

PWLCM are calculated using the keys key_1 and key_2 and the number of iterations are calculated using the keys key_3 and key_4. Thus, the total key space is 2^{128}, Thus, on an average 2^{127} operations are required to perform the brute force attack. since this accounts a very large number, the proposed system can withstand brute force attack. Key space can be further increased by keeping the parameters μ, T and K as secret.

4.2 Known plain text attack

In known plaintext attack, the known pairs of plainttext and ciphertext are used for identifying the remaining plaintexts. In the proposed approach, the PWLCM is run for random number of iterations to generate random binary sequence which is used to initialise the LFSR that generates the random elements of measurement matrix used for CS operation. The PWLCM is again run for random number of iterations to generate another random binary sequence which is then used to construct Arnold matrix for performing the scrambling operation. Each elements in the measurement matrix and the Arnold matrix depends on all the values of the random sequence generated by PWLCM, and hence the probability to retrieve them without knowing the key is very less. The use of CS avoids the transmission of plaintext and the scrambling of compressively sensed vector breaks the relation between the plaintexts and corresponding ciphertexts since it changes the order of elements in the compressed vector. Even though an attacker knows the plaintext and corresponding ciphertext of a block, he cannot find the plaintext-ciphertext relation due to the effect of scrambling. Hence, the proposed system is highly stable against known plaintext attack. An encryption system which can withstand known plaintext attack can withstand chosen ciphertext only attack, where the complete data retrieval is performed from a number of chosen ciphertexts, since among various types of attacks the known plaintext attack has the maximum information about the original data.

5 Experimental results

Experimental results show that the proposed scheme guarantees highly secure audio communication. The scheme considerably reduces the transmission load and breaks the correlation between audio samples effectively in addition to being robust to data loss attacks. This algorithm is applicable to speech and music audio files having different sizes.

The performance of the proposed encryption scheme is evaluated from five perspectives:

1 the reconstruction quality

2 perceptual evaluation of speech quality (PESQ)

3 SD

4 cross-correlation coefficient

5 resistance to noise.

We tested several audio files for different values of sub-rate S by varying it from 0.1 to 0.5. In Section 5.1, the reconstruction quality test results and discussions are presented. Section 5.2 verifies PESQ. SD test results are presented in Section 5.3. Correlation analysis results are drawn in section 5.4 and the result of verification of the robustness to noise is given in Section 5.5.

5.1 The reconstruction quality

The reconstruction quality can be measured by calculating signal to noise ratio (SNR). It is defined as the ratio between original signal power to the noise power. Noise is termed as the difference between the original audio sample values and the reconstructed sample values. SNR can be calculated as

$$SNR = 10\log_{10}\frac{X^2}{\left(X - X_{rec}\right)^2} \tag{13}$$

where X_{rec} is the reconstructed audio file.

SNR of different files, reconstructed by using l_1-minimisation and OMP, after scrambling and descrambling, for different sub-rate is shown in Table 1. From these results, it is clear that the reconstruction quality of both algorithms is good. At lower values of sub-rate S, OMP is performing better than l_1-minimisation. But as S increases, performance of l_1-minimisation is superior to that of OMP. In all cases, SNR increases with S.

Table 1 SNR values of different files for various sub-rate, S (incorrect reconstruction)

Sl. no.	*File*	*S = 0.1*	*S = 0.2*	*S = 0.3*	*S = 0.4*	*S = 0.5*
			l_1 minimisation			
1	Music	8.0104	11.9917	15.4787	18.6729	22.1571
2	Speech	6.3885	11.6741	16.2038	19.9260	23.3911
3	Guitar	7.9383	13.9967	19.0281	23.0887	26.0645
4	Voice	2.6756	7.02010	12.3697	20.0540	33.7731
5	Mix	3.0673	6.25460	9.07000	11.7339	14.5620
			OMP			
1	Music	8.7317	12.5764	15.1706	18.4073	20.1321
2	Speech	6.1047	11.5603	14.6702	16.5027	18.4355
3	Guitar	8.5241	14.7101	19.1001	21.8247	24.4125
4	Voice	2.6669	6.4427	10.9147	14.3733	17.7757
5	Mix	3.7840	6.5032	8.6127	11.1190	12.9126

The security of the proposed algorithm can be verified by reconstructing the audio signal and then calculating SNR using this reconstructed signal under the following two conditions:

1 using the scrambled file, i.e., without performing descrambling

2 using a wrong measurement matrix.

The result obtained in the above mentioned conditions is shown in Tables 2 and 3. SNR is very poor for all values of S in both conditions and hence it is concluded that the proposed method offers security to the data.

The following plots show the effectiveness of the proposed algorithm. Original signal is displayed in Figure 2. Reconstructed signal using scrambling and descrambling is shown in Figure 3. Signal reconstructed using the scrambled signal, i.e., without descrambling is shown in Figure 4. Reconstruction is performed in both cases for a sub-rate of 0.3. These plots clearly indicate that the proposed algorithm offers nearly perfect reconstruction in first case and a poor reconstruction in second case.

Table 2 SNR values of different files for various sub-rate, S (without descrambling)

Sl. no.	*File*	*S = 0.1*	*S = 0.2*	*S = 0.3*	*S = 0.4*	*S = 0.5*
			L_1 minimisation			
1	Music	–1.4787	–1.8100	–2.0463	–2.2347	–2.4043
2	Speech	–1.4993	–1.8385	–2.0561	–2.2163	–2.4191
3	Guitar	–1.4921	–1.7984	–2.0616	–2.2369	–2.4369
4	Voice	–1.4577	–1.8037	–2.0426	–2.2464	–2.3616
5	Mix	–1.4713	–1.8124	–2.0686	–2.2413	–2.4413
			OMP			
1	Music	–2.1763	–2.3719	–2.5232	–2.7681	–2.9275
2	Speech	-1.3249	–1.7563	–2.0212	–2.1891	–2.3986
3	Guitar	–2.2863	–2.8209	–2.9867	–2.9912	–3.0483
4	Voice	–2.2811	–2.9743	–3.0068	–3.1209	–3.2361
5	Mix	–2.3201	–2.8474	–2.9845	–3.00091	–3.1088

Table 3 SNR values of different files for various sub-rate, S (using wrong Φ)

Sl. no.	*File*	*S = 0.1*	*S = 0.2*	*S = 0.3*	*S = 0.4*	*S = 0.5*
			l_1 minimisation			
1	Music	–1.4441	–1.8211	–2.0557	–2.2516	–2.3904
2	Speech	–1.5247	–1.7673	–2.0494	–2.2081	–2.4006
3	Guitar	–1.4955	–1.8340	–2.0077	–2.2311	–2.3952
4	Voice	–1.5342	–1.7805	–2.0473	–2.2010	–2.4349
5	Mix	–1.4841	–1.8129	–2.0477	–2.2413	–2.3952
			OMP			
1	Music	–2.2806	–2.8971	–2.9122	–2.9347	–2.9901
2	Speech	–1.4993	–1.8385	–2.0561	–2.2163	–2.4191
3	Guitar	–2.2727	–2.8677	–2.9781	–2.9963	–3.0812
4	Voice	–2.3832	–2.9927	–3.0281	–3.1464	–3.3616
5	Mix	–2.3326	–2.9147	–2.9968	–3.0211	–3.1134

Figure 2 Original signal (see online version for colours)

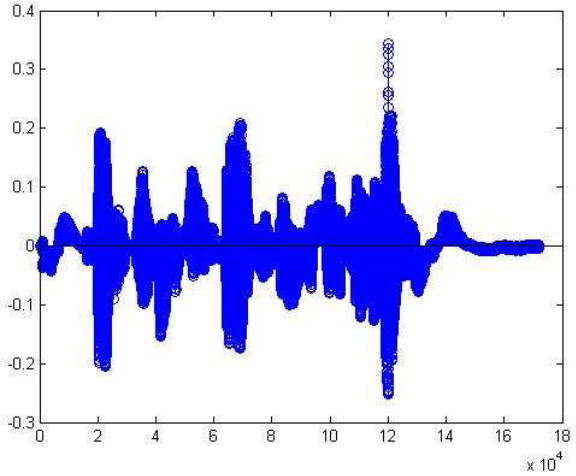
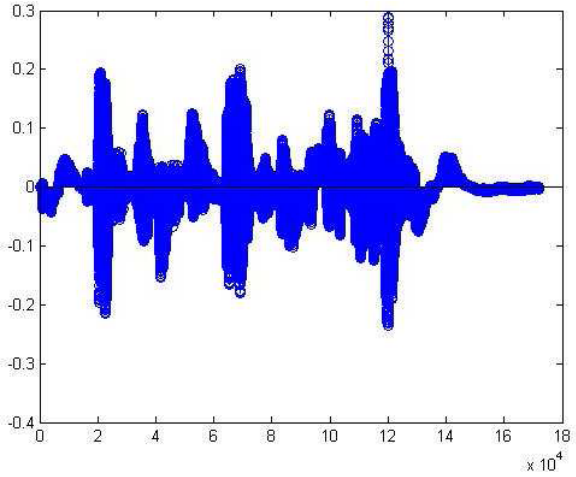


Figure 3 Reconstructed signal after scrambling and descrambling (see online version for colours)

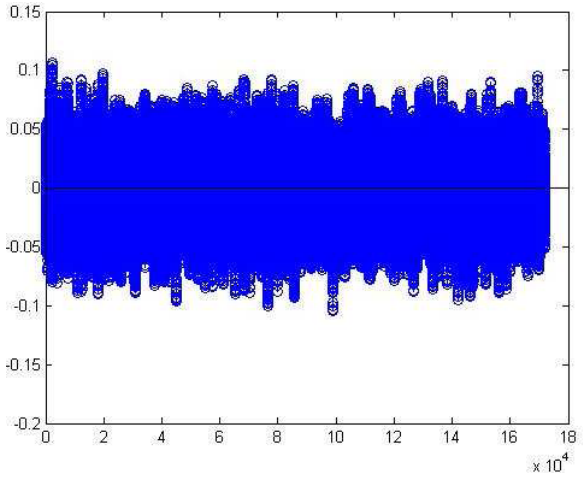


5.2 Perceptual evaluation of speech quality

The PESQ is an international standard designed to predict subjective mean opinion score (MOS) of a degraded audio sample. PESQ returns a score from 4.5 to –0.5, with higher scores indicating better quality. PESQ is designed to analyse specific parameters of audio, including time warping, variable delays, transcoding, and noise. It is primarily intended for applications in codec evaluation and network testing.

PESQ uses a perceptual model to convert the original and degraded speech into an internal representation. The degraded speech is time aligned with the original signal to compensate for the delay that may be associated with the degradation. The difference in the internal representations between the two signals is then used by the cognitive model to estimate the MOS.

Figure 4 Reconstructed signal after scrambling only (see online version for colours)



PSEQ values of different files calculated after reconstructing using l_1-minimisation, for different sub-rates, S is shown in Table 4. From these results, it is clear that the PESQ is good especially for higher subrates, S.

Table 4 PESQ values of different files for various sub-rate, S

Sl. no.	*File*	*S = 0.1*	*S = 0.2*	*S = 0.3*	*S = 0.4*	*S = 0.5*
1	Music	1.6942	1.9732	2.5643	2.9269	3.7192
2	Voice	1.0087	1.5327	2.2976	3.5232	4.2619
3	Speech	1.5676	1.8216	2.5765	3.3115	3.8961
4	Mix	1.2457	1.5693	1.8607	1.9388	2.4016
5	Guitar	1.6294	2.3917	3.2443	3.7266	3.9916

5.3 The SD

SD (Madain et al., 2012) is a measure used to indicate the performance of scrambling algorithm. It can be calculated as follows.

Let $P(i)$ be the original audio sample and L is the length of the audio file, then the difference D for i^{th} cell is calculated as follows:

$$D(i) = \frac{1}{4}\sum_{i'} P(i) - P(i') \tag{14}$$

where $(i') = [(i-1), (i-2), (i+1), (i+2)]$.

Then the mean difference M for the audio file is calculated as

$$M = \frac{\sum_{i=3}^{L-2} D(i)}{L-4} \tag{15}$$

The SD is defined as

$$SD = \frac{M' - M}{M' + M} \tag{16}$$

where M' is the mean difference of the scrambled file and M is the mean difference of the original audio file. The value of SD ranges from –1 to 1. Higher value of SD indicate better scrambling.

The SD for different values of sub-rate S is evaluated by using different files of different size. The result obtained is shown in Table 5. From the results, it is evident that the proposed method guarantees excellent scrambling performance. As sub-rate S increases SD also increases and approaches its maximum value.

Table 5 SD values of different files for various sub-rate, S

Sl. no.	*File*	*S = 0.1*	*S = 0.2*	*S = 0.3*	*S = 0.4*	*S = 0.5*
1	Music	0.8953	0.9186	0.9272	0.9366	0.9421
2	Voice	0.7567	0.7983	0.8250	0.8427	0.8577
3	Speech	0.8977	0.9135	0.9282	0.9343	0.9408
4	Mix	0.8799	0.9106	0.9132	0.9156	0.9184
5	Guitar	0.9353	0.9511	0.9559	0.9608	0.9636

5.4 Correlation coefficient

Correlation coefficient, denoted by ρ is a measure of similarity of two waveforms, giving a value between +1 and 1 inclusive, where 1 is total positive correlation, 0 is no correlation, and 1 is negative correlation. It is widely used as a measure of the degree of linear dependence between two variables and is defined as the covariance of the two waveforms divided by the product of their standard deviations. Mathematically, it can be expressed as

$$\rho = \frac{\mathrm{cov}(X, Xrec)}{\sigma_X \sigma_{Xrec}} \tag{17}$$

where cov(X, $Xrec$) is the covariance of original and reconstructed audio files and σ_X and σ_{Xrec} are the standard deviations of original and reconstructed audio files respectively. The results of correlation analysis of original audio signal to the reconstructed signal are given in Tables 6 and 7. Nearly perfect correlation is obtained for all the files, if it is reconstructed after performing both scrambling and descrambling. The files reconstructed using scrambled files shows poor correlation to the original files. From this analysis, it is evident that this scrambling scheme breaks the correlation between audio files excellently.

Table 6 Correlation coefficient of different files (correct reconstruction) for various sub-rate, S

Sl. no.	*File*	*S = 0.1*	*S = 0.2*	*S = 0.3*	*S = 0.4*	*S = 0.5*
1	Music	0.9327	0.9735	0.9865	0.9923	0.9953
2	Voice	0.6291	0.8795	0.9582	0.9819	0.9918
3	Speech	0.8703	0.9636	0.9836	0.9933	0.9954
4	Mix	0.7632	0.8829	0.9335	0.9602	0.9745
5	Guitar	0.9244	0.9827	0.9937	0.9968	0.9981

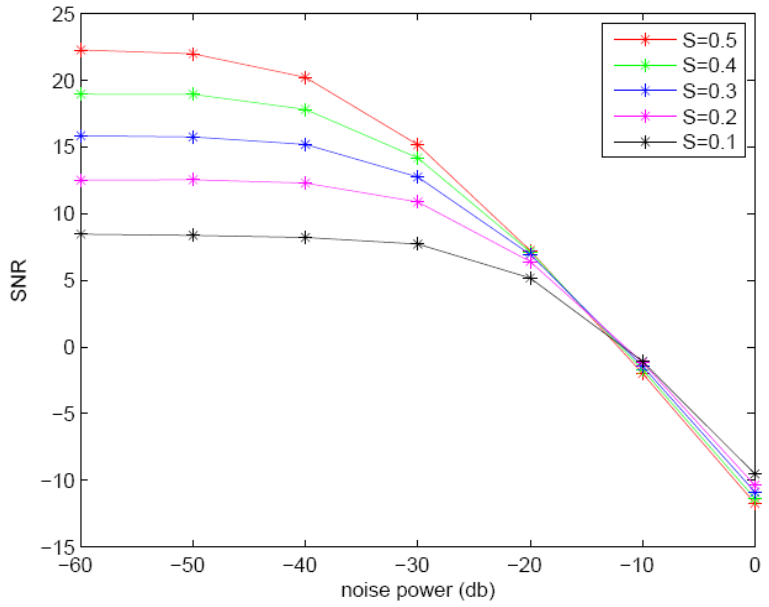
Table 7 Correlation coefficient of different files (reconstructed using scrambled file) for various sub-rate, *S*

Sl. no.	*File*	*S = 0.1*	*S = 0.2*	*S = 0.3*	*S = 0.4*	*S = 0.5*
1	Music	0.0007	0.0026	0.0013	–0.0007	–0.0014
2	Voice	0.0009	–0.0038	–0.0062	–0.0011	–0.0008
3	Speech	–0.0003	0.0023	–0.0035	0.0077	–0.0042
4	Mix	0.0002	–0.0005	–0.0028	–0.0016	–0.0009
5	Guitar	0.0021	0.0020	–0.0001	0.0015	0.0011

5.5 *Resistance to noise*

Robustness in the presence of noise is tested by adding a white Gaussian noise with the scrambled audio signal and the resultant signal is used for reconstruction. The noise power is varied from –60 dB to 0 dB. The results obtained for audio file *music* using l_1-minimisation and OMP reconstruction methods are shown in Figure 5 and Figure 6, respectively. Up to a certain value of noise power SNR remains almost constant and then it decreases rapidly. As sub-rate increases SNR also increases. From the graph, it is evident that, the proposed method guarantees a satisfactory reconstruction performance up to a noise power of –20 dB.

Figure 5 SNR for different values of *S* for different values of noise power (l_1-minimisation) (see online version for colours)

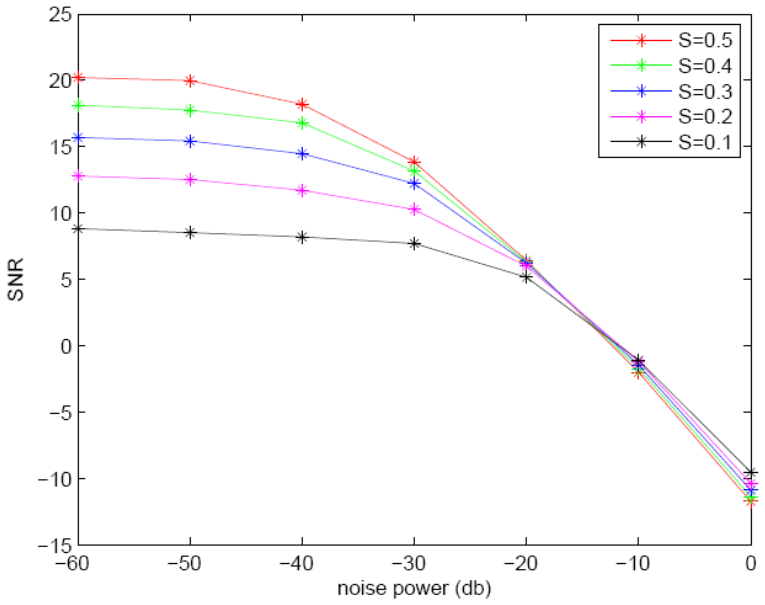


6 Comparison with other schemes

6.1 Comparison of attack complexity

A key-based measurement matrix construction procedure for CS using LFSR is proposed in George and Pattathil (2014). Let the length of LFSR be equal to 16. Here, the key is the initial seed of LFSR whose length is also 16. Thus, the key space is 2^{16} and 2^{15} operations are required to perform the brute force attack. But in the encryption scheme proposed in this paper, even though the measurement matrix is constructed by LFSR of same length, the use of PWLCM and three 32-bit keys increases the key space to 2^{96}. Moreover, the Arnold matrix used for scrambling is also created by using another 32-bit key. So the total key space is 2^{128} and 2^{127} operations are required to break the key. The Arnold matrix generation proposed in Huang et al. (2012) uses a 64-bit key and hence the key space is only 2^{64}. The key space of audio scrambling scheme proposed in Augustine et al. (2014) is 2^{96} which is also less than that of proposed method. Due to this enlarged key space the attack complexity of the proposed encryption method is superior to that of existing schemes.

Figure 6 SNR for different values of *S* for different values of noise power (OMP) (see online version for colours)



6.2 Comparison of computational complexity

The aim of CS is to unify sampling and compression operations and it reduces the data acquisition and computational load at the encoder at the cost of increased computation at the intended receiver. Thus CS considerably reduces the encoder complexity and storage requirement. A speech encryption scheme based on scrambling via CS is proposed in Zeng et al. (2012). This scheme is computationally complex and necessitates additional

storage requirements. A sparsification is required before taking random measurements of speech frames. This introduces computational overheads and violates the basic principle of CS. Here, CS is only employed as a dimension reduction method. The use of stochastic dictionary introduces additional storage requirement at both encoder and decoder side. The predecoding delays the encryption process and makes the encoder too complex since this introduces all complexities of decoder into the encoder also, which is against the concept of CS. Moreover, the scrambling/desrambling method also requires a lot of computation to find the scrambling/descrambling matrix and encrypted vector and to separate one frame from the encrypted vector.

In the proposed scheme, no sparsification and prereconstruction operations are required and the use of stochastic matrix dictionary is also not required. The random measurements are taken by using a key-based stochastic matrix constructed by using secret keys, PWLCM and LFSR. This matrix satisfies RIP and incoherence and offers security in addition to dimension reduction. The scrambling procedure is simple and requires only a few multiplications and divisions.

Hence, our scheme offers significant reduction in the computational complexity and considerable improvements in speed of operation when compared to the encryption method proposed in Zeng et al. (2012).

7 Conclusions

A new encryption technique for digital audio signal has been introduced. The proposed scheme takes the advantages of key-based secure CS and Arnold transform to achieve high security, excellent compression, robustness and good SD. This paper studies the effect of variation in sub-rate on reconstruction quality, PESQ, SD and robustness to noise. The method can be used for audio files of different size and characteristics. Experimental results show that the scheme is very efficient.

References

Augustine, N., George, S.N. and Deepthi, P. (2014) 'Compressive sensing based audio scrambling using Arnold transform', in *Recent Trends in Computer Networks and Distributed Systems Security*, Springer, pp.172–183.

Candès, E.J. and Wakin, M.B. (2008) 'An introduction to compressive sampling', *Signal Processing Magazine*, Vol. 25, No. 2, pp.21–30, IEEE.

Del Re, E., Fantacci, R. and Maffucci, D. (1989) 'A new speech signal scrambling method for secure communications: theory, implementation, and security evaluation', *IEEE Journal on Selected Areas in Communications*, Vol. 7, No. 4, pp.474–480.

Donoho, D.L. (2006) 'Compressed sensing', *IEEE Transactions on Information Theory*, Vol. 52, No. 4, pp.1289–1306.

George, S.N. and Pattathil, D.P. (2014) 'A secure LFSR based random measurement matrix for compressive sensing', *Sensing and Imaging*, Vol. 15, No. 1, pp.1–29.

Huang, R., Rhee, K. and Uchida, S. (2012) 'A parallel image encryption method based on compressive sensing', *Multimedia Tools and Applications*, Vol. 72, No. 1, pp.1–23.

Krawczyk, H. (1994) 'LFSR-based hashing and authentication', in *Advances in Cryptology CRYPTO94*, Springer, pp.129–139.

Li, H., Qin, Z., Zhang, X. and Shao, L. (2010) 'An n-dimensional space audio scrambling algorithm based on random matrix', *Journal of Xi'an Jiaotong University*, Vol. 4, p.5.

Lin, Y. and Abdulla, W. (2007) 'A secure and robust audio watermarking scheme using multiple scrambling and adaptive synchronization', in *Information, Communications & Signal Processing, 2007 6th International Conference on IEEE*, pp.–5.

Madain, A., Dalhoum, A.L.A., Hiary, H., Ortega, A. and Alfonseca, M. (2012) 'Audio scrambling technique based on cellular automata', *Multimedia Tools and Applications*, pp.1–20.

Nan, L., Yanhong, S. and Jiancheng, Z. (2004) 'An audio scrambling method based on Fibonacci transformation', *J. North China Univ. Technol*., Vol. 16, No. 3, pp.8–11.

Orsdemir, A., Altun, H.O., Sharma, G. and Bocko, M.F. (2008) 'On the security and robustness of encryption via compressed sensing', in *Military Communications Conference, 2008, MILCOM 2008*, IEEE, pp.1–7.

Satti, M. and Kak, S. (2009) 'Multilevel indexed quasigroup encryption for data and speech', *IEEE Transactions on Broadcasting*, Vol. 55, No. 2, pp.270–281.

Senk, V., Delic, V. and Milosevic, V. (1997) 'A new speech scrambling concept based on Hadamard matrices', *Signal Processing Letters*, Vol. 4, No. 6, pp.161–163, IEEE.

Servetti, A. and De Martin, J.C. (2002) 'Perception-based partial encryption of compressed speech', *IEEE Transactions on Speech and Audio Processing*, Vol. 10, No. 8, pp.637–643.

Shang, Z., Ren, H. and Zhang, J. (2008) 'A block location scrambling algorithm of digital image based on Arnold transformation', in *The 9th International Conference for Young Computer Scientists, 2008, ICYCS 2008*, IEEE, pp.2942–2947.

Tropp, J.A. (2004) 'Greed is good: algorithmic results for sparse approximation', *IEEE Transactions on Information Theory*, Vol. 50, No. 10, pp.2231–2242.

Zeng, L., Zhang, X., Chen, L., Fan, Z. and Wang, Y. (2012) 'Scrambling-based speech encryption via compressed sensing', *EURASIP Journal on Advances in Signal Processing*, Vol. 2012, No. 1, pp.1–12.