Frequency Speech Scrambler based on Hartley Transform and OFDM Algorithm

Anjana D S
Department of Electronics and Communication
Federal Institute of Science and Technology
Ernakulum, India

Minu Kuriakose
Assistant Professor
Department of Electronics and Communication
Federal Institute of Science and Technology
Ernakulum, India

ABSTRACT

Speech scramblers are considered for communication applications where secured speech signal transmissions are important requirement. This paper proposes a frequency voice scrambler with much less residual intelligibility, simulated in Matlab. In this, two techniques, Fast Hartley Transform (FHT) and Orthogonal Frequency Division Multiplexing (OFDM) scrambling key generation scheme are combined together, which can provide scrambled speech by permuting frequency components. Thus the voice cryptographic system is based on an OFDM scrambled discrete Hartley Transform.

General Terms

Scrambling, Permutation, Cryptography, Algorithm.

Keywords

Fast Hartley Transform (FHT), OFDM scrambling key generation algorithm, encoder, decoder.

1. INTRODUCTION

There has been an expected rapidly growing interest in, and development of, secure communication techniques in relation to the activities of military services, banking systems and other systems where degree of secured speech signal transmission plays a major role. Scrambling is used to keep the secrecy of speech signal over unauthorized listeners. It is simply disordering of the speech signal so that it is no longer intelligible. The original speech signal can be recovered by the intended receiver through appropriate descrambling technique. Among speech scramblers, analog speech scramblers are considered due to their wide applicability.

The scrambling techniques could be classified as time-domain and frequency-domain scrambling. In time-domain scrambling, speech signals are divided into small time interval units and these units are permuted [1-3].As these units could be as small as just one sample, scrambling results in bandwidth expansion. This can lead to loss of signal out of band of the channel and thereby degrading the speech quality.

In frequency-domain scrambling, speech signals are separated into several subbands and these subbands are then permuted. It ensures the original bandwidth is kept unchanged. In the frequency- domain, the first algorithms used were based on Fast Fourier Transform (FFT) technique, where the FFT coefficients are permuted frame to frame [4-6]. Techniques based on Discrete Cosine Transform [7], Hadamard Transform [8], Wavelet Transform [9], Principal Component Analysis [10] etc have also been subject of studies.

Voice cryptography system based on Fast Hartley Transform (FHT) is a recent development. Nascimento and Toscano [11] used pseudorandom permutation of frequency components in FHT with insertion of random frequency components as scrambling technique.

Now-a-days the main focus is also given to the scrambling or permutation algorithms which provide efficiently scrambled unintelligible speech signals. Earlier Tompkins-Paige algorithm [4] was used to generate permutation. Then conventional permutations were considered like in [7] and [9]. Later Raymond and Cyril [6] proposed a derangement generation algorithm for key generation. A new algorithm, OFDM scrambling key generation algorithm, introduced by Tseng and Chiu [12] is used here for permutation of frequency components in FHT. In this work each Hartley coefficient becomes frequency subcarrier and also the OFDM scrambling key generation algorithm used provides scrambled signal with much less intelligibility.

Section 2 presents Fast Hartley Transform (FHT) which is one of the important modules. Section 3 explains another important section of the proposed method, i.e. about OFDM scrambling key generation algorithm. Section 4 and 5 describe the whole encoder and decoder section of the scrambling system. The simulated results are given in the Section 6 and the last section provides conclusions of the work.

2. FAST HARTLEY TRANSFORM (FHT)

The discrete Hartley transform converts sequence of real numbers of size N in the time domain into sequence of real numbers of size N in frequency domain [13]. The discrete Hartley transform implemented in fast algorithm called Fast Hartley Transform (FHT) has the computational complexity of N $\log_2$ (N). The pair of the discrete Hartley transforms may be written as:

$$H[k] = \sqrt{\frac{1}{N}} \sum_{n=0}^{N-1} x[n] \text{cas}\,(\frac{2\pi}{N}\,kn\,), 0 \le k \le N-1 \qquad (1)$$

$$x[n] = \sqrt{\frac{1}{N}} \sum_{n=0}^{N-1} H[k] \text{cas}\,(\frac{2\pi}{N}\,kn\,), 0 \le k \le N-1 \qquad (2)$$

Where H[k] represents the Hartley coefficient of index k, x[n] the input real sequence of N size, and **cas(z)=sin(z)+cos(z)** the basis function of the respective transform.

Recently Hartley transform is taken as an alternate to Fourier transform. The main specification of FHT is that it takes real terms rather than complex terms as in FFT and also its inverse is same as the direct transform which makes it a as directionless transform. Hartley transform removes the redundancy in the Fourier domain by repacking the complex coefficients through the relation,

$$\text{H[k]} = \text{Re}\big[\text{X[k]}\big] - \text{Im}\big[\text{X[k]}\big] \qquad (3)$$

2.1 Fast Hartley Transform algorithm

a) Split N- point sequence into two smaller N/2 point sequences x1[n] and x2[n].

b) Time sequence x2[n] is shifted to left by one by the shift rule of Hartley transform.

c) The two sequences x1[n] and x2[n] are then Hartley transformed to H1[k] and H2[k] and combined into H[k].

Using periodic properties, the equation H[k] is given by,

$$H[k] = \begin{cases} H1[k] + cos\big((2\pi/N)k\big)H2[k] + \ sin\big((2\pi/N)k\big)H2[-k] \\ \qquad\qquad 0 \le k \le \frac{N}{2} - 1 \qquad (4) \\ H1[k - N/2] - cos\left((2\pi/N)\left(k - \frac{N}{2}\right)\right)H2[k - N/2] \\ -sin\left((2\pi/N)\left(k - \frac{N}{2}\right)\right)H2[-k + N/2] \ , \quad \frac{N}{2} \le k \le N - 1 \\ \qquad\qquad (5) \end{cases}$$

These equations form the basic formula for Fast Hartley Transform algorithm [14]. Since it requires three values for each k, FHT algorithm takes a double butterfly flow graph to implement the equation.

3. SCRAMBLER DESIGN

3.1 Frame Structure

The frame size of FHT operation in scrambler is fixed to 256 samples. As the frame size increases quality of descrambled speech signal increases, at the same time the processing delay also increases. For the tradeoff between quality and processing delay, the frame size is determined to be 256 samples [4].

3.2 Frequency and Band Width parameter settings

The speech signal is sampled at a rate of 8000 samples per sec. The input signal is processed over a band width of 3.1 kHz, ranges from 300-3400 Hz. The transmission channel band limited to the range of 0-4000Hz.The 256 samples frame size results in 32ms frame time (Table 1).

Table 1. Design Parameters.

FHT frame size (no. of subcarriers)	256 samples(32ms)
Sampling Frequency	8kHz
Input speech signal BW	3.1kHz(300-3400Hz)
Channel BW	4kHz(0-4000Hz)

3.3 OFDM Scrambling Key Generation Algorithm

The proposed scrambling algorithm has 256 subcarriers as the frame size itself is considered as subcarriers. Scrambling is implemented symmetrically in relation to the coefficient of index N/2. I.e. scrambling is applied symmetrically to the bands. Please To eliminate bandwidth expansion, all 93 frequency components corresponding to the subcarriers numbered 12 to 104 (as symmetrically applied, 150 to 243 also) are permuted and rest are set to zero. The system thus has a frequency range of 375-3270 Hz, which is within the original speech signal bandwidth.

Scrambling key generation controlled by two main factors [12]. 1) Seed S0 and 2) a secret key, which is an embedded permutation pattern (I1,I2…..I93). A random number R_i, corresponds to ith position in 1st frame, is first generated from the seed S0. After that the algorithm creates the scrambling key as follows.

Step 1: For the 1st frame, load the embedded permutation pattern.

Step 2: For $i = 93, \ 92, \dots, \ 2$

2A: R_i = a random number uniformly distributed between 0 and 1

2B: k_i = ceil (i * R_i)

2C: exchange I_i and I_k , for next frame these I pattern taken as permutation pattern.

Where ceil(x) denotes the minimum integer which is greater than or equal to x. Scrambling key generated in step 2 is the permutation version of embedded permutation pattern. There are 93! possible scrambling key for frequency components permutation in each half frame. For any permutation P on {1,2…N}, there is always an inverse permutation P^{-1} , such that P^{-1} (P(i))=i.

4. CRYPTOGRAPHIC ENCODER

The functional block diagram of encoder corresponding to the scrambling of speech signal is shown in Fig 3. At the encoder, an analog speech signal is sampled at 8000Hz and digitized by an ADC. The voice signal then passes through an AGC unit, which provides dynamic range conditioning to the signal, in order to avoid overflow in subsequent Fast Hartley transform algorithm processing. Then discrete Hartley transform is applied to a frame size of 256 conditioned voice samples, leading to a vector with 256 coefficients in frequency domain.

The Hartley transformed coefficients are then given to the scrambler which scrambles the coefficients according to the OFDM scrambling keys generated by the algorithm process. After scrambling, inverse Hartley transform is applied to the frequency scrambled Hartley coefficients, which is same as the direct transform. The IFHT output is converted to analog by DAC and after multiplexing with synchronization signal and training sequence it is released to the channel.

5. CRYPTOGRAPHIC DECODER

The functional block diagram of decoder which depicts the descrambling of received speech signal is shown in Fig 4. From the digitized received signal, first sync signal is detected to adapt the channel equalization. The frame synchronization detector has to be prepared to receive resynchronization signal periodically. Then the signal is passed to channel equalizer where the phase distortions occurred in the channel is reduced. Adaptive transverse equalization algorithm is used.

The scrambled and equalized signal is then passed to discrete Hartley transform section, converting it to frequency domain. Inverse permutation is applied to descramble the signal. After that inverse Hartley transform is performed to convert the descrambled signal to time domain. The digitized descrambled signal finally takes its analog form by DAC unit and the speech signal is produced.

Table 2. Comparison with existing method.

Method [11]	Proposed Method
The Hartley coefficients are grouped as 5 coefficients together, making frequency band	Each Hartley coefficient taken as frequency subcarrier
Uses pseudo random scrambling	Uses OFDM scrambling key generation algorithm
Inserts random frequency components, which is to be known at receiver	Uses Embedded permutation pattern with seed, which is to be known at receiver
A total of approximate 25! permutation possible in each half frame	A total of 93! possible scrambling key to permute in each half frame

Table 2 shows that the proposed method provides more efficient voice cryptographic system characteristics compared to the method used in [11].

6. SIMULATION RESULTS

The proposed scrambling method is simulated on a test sample audio, 25 sec part of Senate speech of Barak Obama, using MATLAB 7 (or above). The encoder section takes a total of 5minutes and decoder takes around 6minutes. The results are shown in Fig 1.

Fig.1(a) corresponds to the original speech signal and Fig.1(b) and Fig.1(c) corresponds to the scrambled and descrambled speech signals respectively. The Fig 2 represents the enlarged portion of 100-356 samples of the Fig 1.From the figure it can be seen that the scrambled signal loses the actual characteristics of voice signal and the descrambled signal approximately regains it. When the scrambled speech is heard, one cannot identify the words or even cannot recognize as a speech. The descrambled speech reproduces the words and one can clearly understand it. But the magnitude of speech signal gets changed.

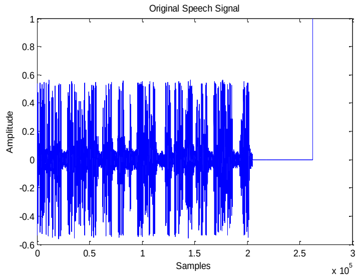

Fig.1(a) Input Voice Signal

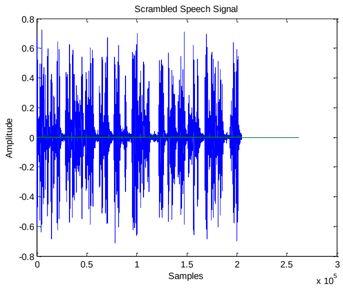

Fig. 1(b) Scrambled Speech Signal

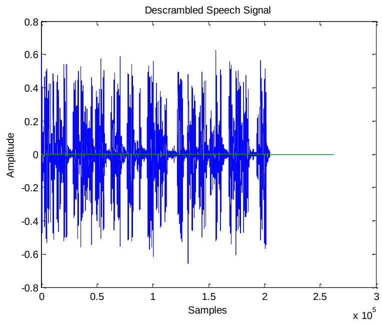

Fig.1(c) Descrambled Speech Signal

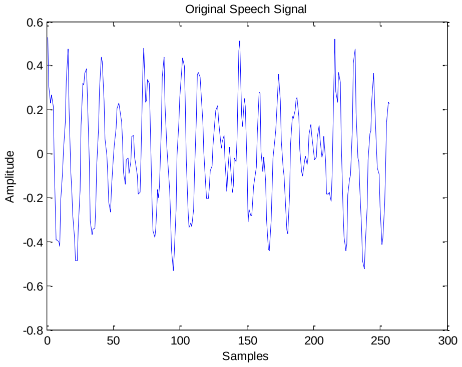

Fig 2(a). Enlarged Input Voice Signal

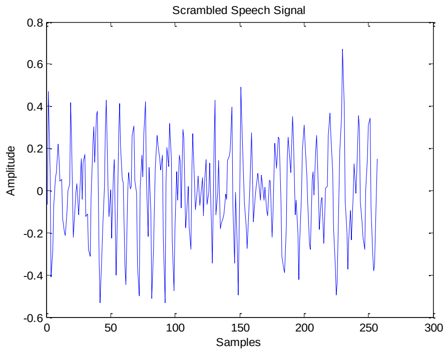

Fig 2 (b). Enlarged Scrambled Signal

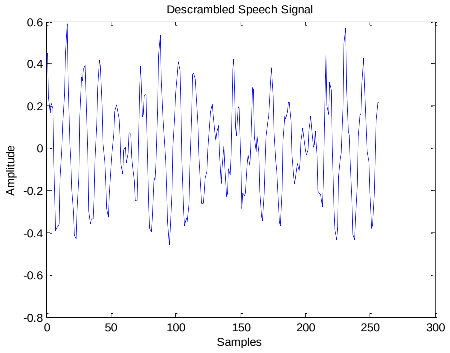

Fig 2 (c).Enlarged Descrambled Signal

7. CONCLUSION

The main objective of proposed method is to create a voice cryptography system which provides security to information, ability to reconfigure internal modules and encryption or scrambling algorithm. The Fast Hartley Transform is used here because it is very much similar to Fast Fourier Transform but only real terms are considered. Also the direct transform of FHT is equal to its inverse. The transform operations takes less time compared to Fourier operations. OFDM scrambling key algorithm provides good encryption of speech signal with much less intelligibility at the output. Also the algorithm resists brute force and frequency domain attacks efficiently, as it provide a total of 93! permutation key space in each half frame, keeping the recovered speech signal at a satisfactory level. It does not suffer from bandwidth expansion problem.

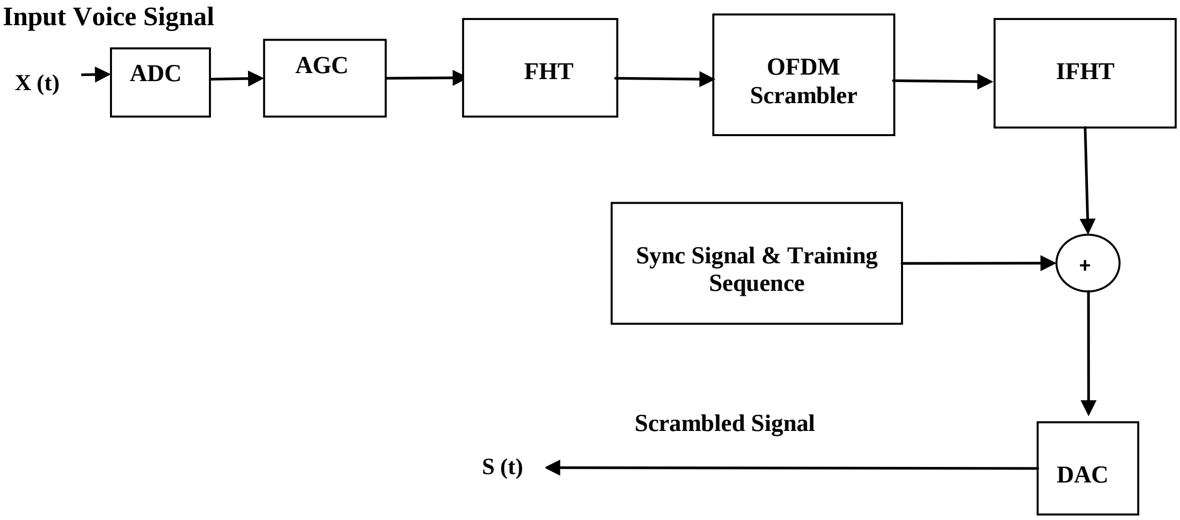

Fig.3 Cryptographic Encoder

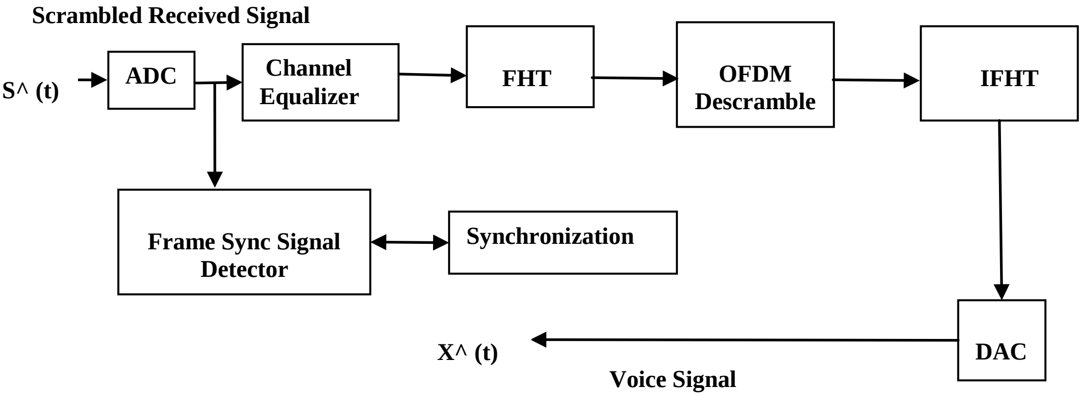

Fig .4 Cryptographic Decoder

8. REFERENCES

[1] Lee, Lin-Shan and Chou, Ger-Chih (1984). A New Time Domain Speech Scrambling System Which Does Not Require Frame Synchronization. IEEE Journal of Selected Areas in Communications.

[2] Mitchell, C.J.; Piper, F.C (1985). A classification of time element speech scramblers. Journal of the Institution of Electronic and Radio Engineers, Vol. 55, pp. 391 - 396.

[3] Huang, F.; Stansfield, E.V. (1993). Time sample speech scrambler which does not require synchronization. IEEE Transactions on Communications, Vol. 41, pp. 1715 - 1722.

[4] Sakurai, K. Koga, K. Muratani, T. (1984). A Speech Scrambler Using the Fast Fourier Transform Technique. IEEE Journal on Selected Areas in Communications, Vol. 2, pp. 432-442.

[5] Ehsani, M.S. and Borujeni, S.E. (2002). Fast Fourier transform speech scrambler. Proceedings of The First International IEEE Symposium on Intelligent Systems, Vol. 1, pp. 248 - 251.

[6] Woo, Raymond W. and Leung, Cyril (1997). A New Key Generation Method for Frequency-Domain Speech Scramblers, IEEE Transactions on Communications, VOL. 45, NO. 7, July 1997, pp. 749-752.

[7] Dawson, E. (1991). Design of a discrete cosine transform based speech scrambler. Electronics Letters, Vol. 27, PP. 613 - 614.

[8] Milosevic, V., Delic, V. and Senk, V. (1997). Hadamard transform application in speech scrambling. 13th International Conference on Digital Signal Processing Proceedings, Vol. 1, pp. 361 - 364.

[9] Ma, Fulong , Cheng, Jun and Wang, Yumin (1996). Wavelet transform-based analogue speech scrambling scheme. Electronics Letters, Vol. 32 , pp.719-721.

[10] Nidaa A. Abbas (2009). Speech Scrambling Based on Principal Component Analysis, Journal of Compunting, Vol. 1, NO. 3, pp. 452-456.

[11] Nascimento, Toscano (2012) Frequency Speech Scrambler Based on the Hartley Transform and the Insertion of Random Frequency Components, the International Journal of Forensic Science Feb 2012.

[12] Tseng, Chiu, An OFDM Speech Scrambler without Residual Intelligibility ,TENCON IEEE REGION 10 Conference 2007 pp 1-5.

[13] Bracewell, R. N. (1983). Discrete Hartley Transform. Journal Opt.Amer., vol. 73, no. 12, pp. 1832-1835.Bracewell, R. N. (1983). Discrete Hartley Transform. Journal Opt.mer., vol. 73, no. 12, pp. 1832-1835..

[14] R.F.Ullman, An Algorithm for the Fast Hartley Transform, in Stanford Exploration Project reports 1984.