# Compressing And Ciphering Digital Signal By Using Wavelet Transform

*Asst. Lecturer* **Emad H. Salman**

**University of Diyala**

## Abstract:

With the rapid development of multimedia and network technologies, the security of multimedia becomes more and more important, since multimedia data are transmitted over open networks more and more frequently. Typically, reliable security in storage and transmission of digital speech data, images, and videos is needed in many real applications, such as medical imaging systems, military image databases, as well as confidential video conferences. In recent years, some consumer electronic devices, such as mobile phones, have also started to provide the function of saving and exchanging digital speech/music data, images, and video clips under the support of multimedia messaging services over wireless networks, which is urgently demanding for multimedia security.

A new proposed method is presented with high degree of security to calculate the transform domain of Discrete Wavelet Transform (DWT) and Inverse Discrete Wavelet Transform (IDWT) by the pyramid algorithm. The pyramid algorithm is permitted to construct the bases of the vectors of its transform, this leads to more security in the system. Also by the pyramid algorithm, the transformation process gives double the number of samples of the original signal, while in any transformation the process gives square the number of samples of the original signal. This work includes a mathematical presentation the modification of DWT (with IDWT) followed by a second stage of DWT (multi resolution) to compress the signal. Thus, the system ciphers the signal by two dimensions (time and frequency) then compresses and ciphers it again. The keys used throughout this paper are generated manually; the first algorithm is for time domain, the second algorithm is for frequency domain and the last algorithm is for compressing signal samples.

Finally, all the preceding methods give very inapprehensible compressed and ciphered signal and intelligible decompressed and deciphered signal according to numerical and graphical results, so this work gives 50% compression ratio and approaches 10% error ratio.

Key words: Signal Compression, Signal Ciphering, Key Management, Time Domain, Frequency Domain, and DWT Coefficients Ciphering, Daubechies DWT multi-resolution, Daubechies DWT single-resolution.

## 1. Introduction

The primary purpose of early computers is computing. With the advancements in technology, computers were developed to provide a wide variety of applications. One of these is the representation and storage of non-numeric information into binary code, e.g. audio signals and images. Such applications require longer data representations for quality. These data requires larger storage space and longer access time. Another setback would be in terms of data transmission. Transmission time would be longer since the data that would be sent is larger. With these disadvantages comes the need for the compression of data [1].

The scrambling methods are considered as important methods that provide the communication systems (based mainly on using a speech signal as an information carrier) with a specified degree of security, depending on the used technique to implement the scrambling method. There are many traditional scrambling methods, some are used in single dimension such as time or frequency domain scrambling, and others are used in more than one dimension method. Among speech scramblers, digital scramblers are attractive due to their wide applicability [2]. Secrecy is certainly important to the security or integrity of information transmission. Indeed, the need for secure communications is more profound than ever, recognizing that the conduct of much of commerce, business, and personal matters is carried out today through many mediums [3]. Speech signal contains two types of information, the content of the speech and the personality of the speaker. Ideally, an encrypted or scrambled voice should conceal both types of information. Speech security devices of switched telephone networks and mobile telecommunication systems fall into two categories, namely, analog speech scramblers and digital speech encryptors [4, 5].

Signal scrambling seeks to perform a completely reversible operation on a portion of signal, such that it is very unintelligible to unauthorized listener. The three most important criteria used to evaluate speech scramblers are [2]:

1. The scrambler's ability to produce encrypted signal with low residual intelligibility.
2. The extent to which the encryption and the decryption processes affect the quality of the signal recovered by intended reception.
3. The scrambler's immunity to cryptanalysis attack.

The Discrete Wavelet Transform (DWT) is used in a variety of signal processing applications, such as video compression, internet communications compression, object recognition, and numerical analysis. It can efficiently represent some signals, especially ones that have localized changes. Consider the example of representing a unit impulse function with the Fourier transform, which needs an infinite amount of terms because we are trying to represent a single quick change with a sum of sinusoids. However, the wavelet transform can represent this short-term signal with only a few terms.

This transform came about from different fields, including mathematics, physics, and image processing. Essentially, people in different areas were doing the same thing, but using different terminology. In the late 1980s, Stéphane Mallat unified the work into one topic [6].

## 2. The Proposed Algorithm

Any signal-scrambling algorithm needs to satisfy the following requirements [7]:

1. The scrambled signal should be unintelligible.
2. The scrambled signal should occupy the same bandwidth, as does the original signal. That is, the scrambling process should be a bandwidth preserving operation.
3. It should be difficult to decrypt, if the decryption key is not available. In other words, it should be cryptanalytically strong or secure.
4. The communication delay caused by the scrambling process must be as small as possible.
5. The recovered signal at the receiver end should be of good quality and should preserve both the intelligibility of the signal and the characteristics of the speaker.

The proposed algorithm is shown in Figure 1; the digital encryptor system will have three major parts, where these three parts are connected in cascade, which are a frequency domain scrambler, time domain scrambler, and DWT coefficients scrambler respectively. It is clear that a frequency scrambling part is before time scrambling part. This causes that an eavesdropper will be distributed over the duration of the delay line.

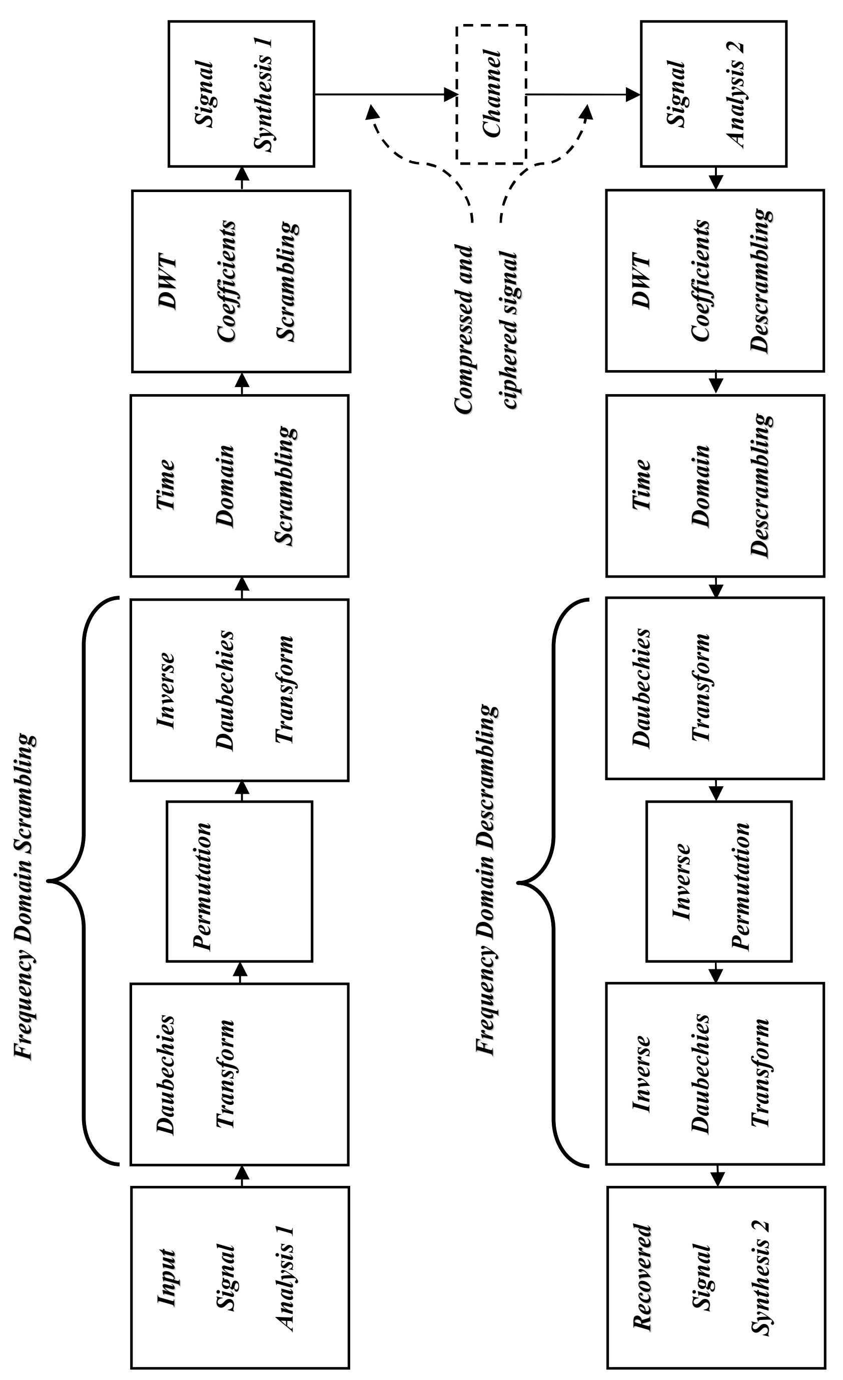
Input Signal Analysis 1
Daubechies Transform
Permutation
Inverse Daubechies Transform
Time Domain Scrambling
DWT Coefficients Scrambling
Signal Synthesis 1
Frequency Domain Scrambling
Channel
Compressed and ciphered signal
Signal Analysis 2
DWT Coefficients Descrambling
Time Domain Descrambling
Daubechies Transform
Inverse Permutation
Inverse Daubechies Transform
Recovered Signal Synthesis 2
Frequency Domain Descrambling

Figure 1: Block diagram of the proposed algorithm.

**The block diagram of Figure 1 consists of the following parts:**

- ***Input Signal Analysis 1:*** **Sampling the input signal according to the sampling frequency then segmenting it into frames according to the segment length.**

- *Daubechies Transform:* Applying the Daubechies wavelet transform for each segment to provide new samples.
- *Permutation:* Computing the frequency domain key according to the segment length to produce a new samples arrangement in each segment.
- *Inverse Daubechies Transform:* Taking the inverse Daubechies wavelet transform for each segment to provide new samples in the time domain.
- *Time Domain Scrambling:* Computing the time domain key according to segments number to produce new segments arrangement.
- *DWT Coefficients Scrambling:* Computing the DWT coefficients key according to half the signal length then taking the DWT coefficients to compress the signal and rearranging these coefficients by their key.
- *Signal synthesis 1:* Synthesize the signal by reassembling and transmitting the scrambled signal.
- *Signal Analysis 2:* Sampling then segmenting the scrambled signal.
- *DWT Coefficients Descrambling:* Applying the DWT coefficients key to recover original arrangement (decompressing the signal from these coefficients).
- *Time Domain Descrambling:* Applying the time domain key to recover original segments arrangement.
- *Daubechies Transform:* Taking the Daubechies wavelet transform for each segment.
- *Inverse Permutation:* Applying the frequency domain key to recover the original samples arrangement.
- *Inverse Daubechies Transform:* Taking the inverse Daubechies wavelet transform for each segment to provide the original samples in the time domain.

*Recovered Signal synthesis 2:* Reassembling the segments to provide the recovered signal.

## 3. Frequency Scrambling

The digital encryption processes that employ a transformation of the input signal to facilitate encryption can best be described using matrix algebra. Consider the vector *x* which contains *N* sampled signals (in time domain) obtained from ADC process (*N* represents a frame of the original signal). Let this sampled signals vector *x* be subject to an *N×N*

Orthogonal transformation matrix *F* such that

$$X(k) = F.x(n) \qquad \ldots (1)$$

This transformation results in a new vector *X(k)* made up of *N* transform coefficients. The *N×N* permutation matrix is then applied to *X(k)* such that each transform coefficient is moved to a new position within the vector. Then

$$Y(k) = P.X(k) \qquad \ldots (2)$$

**A scrambled signal vector *y(n)* is obtained by transforming vector *Y(k)* to the time domain using the inverse transformation $F^{-1}$ where**

$$y(n) = F^{-1}.Y(k) \qquad \dots (3)$$

**Decryption or recovery of the original speech vector *x(n)* is achieved by first transforming *y(n)* to the transform domain. Then the inverse permutation matrix $P^{-1}$ is used to move the transform coefficients to their original position.**

**Finally, the resulting transform vector is transformed to the time domain by multiplying by $F^{-1}$**

$$x'(n) = F^{-1}.P^{-1}.F.y(n) \qquad \dots (4)$$

**The transform domain scrambling process outline above requires the transform matrix *F* to have an inverse. The scrambling transformation $T=F^{-1}.P.F$ must be orthogonal since orthogonal transformations are norm preserving. The inverse transformation $T^{-1}$ will also be orthogonal. This property is useful since the descrambling process as shown below will not enhance any noise added to the scrambled signal during transmission. The scrambled speech sequence is given by**

$$y(n) = F^{-1}.P.F.x(n) = T.x(n) \qquad \dots (5)$$

**At most, $N$ elements can be permuted in the transform-based scrambling process. It is important to note that for a given sampling frequency, $N$ will determine the delay introduced by the scrambling device. Therefore, a tradeoff is made between system delay and security. Usually $N$ is chosen to be equal to 256. Practically, there are $N!$ possible coefficient arrangements. This restriction stems from the requirement that the scrambled speech should occupy the same bandwidth as the original speech. If the transformed components have a frequency representation, those lying outside the allowable band are set to zero and the others are permuted.**

**At the transmitter, a signal is sampled and segmented into frames, then it is transformed into wavelet space, after that the wavelet coefficients are permuted by using good permutations. For example if $G$ is considered as set of permutations then $G$ should satisfy the following requirements:**

- **Any permutation in $G$ must not produce an intelligible scrambled signal.**
- **Any permutation in $G^{-1}$ must not produce an intelligible descrambled signal if the inverse permutation dose not corresponds to the permutation used in the transmitter.**

**After applying permutations, inverse wavelet transform is applied, yielding the scrambled signal. At the receiver, the scrambled signal is transformed back into the wavelet space, then inverse permutation is applied and finally it is transformed using inverse wavelet transform. Then the original signal is produced [2].**

## 4. Wavelet Transform

**The wavelet transform (WT) have many types, the Haar wavelet transform type is quite pleasant to devise a basis that includes the scaling vector (1, 1, 1, 1). The second basis vector can be (1, 1,–1,–1), which is a square wavelet, up once and down once. The German mathematician Alfred Haar completed a perpendicular basis in 1910 by squeezing and shifting. He squeezed (1, 1,–1,–1)**

to produce the third vector (1,–1, 0, 0). Then, he shifted the third vector by two time intervals to produce the forth vector (0, 0, 1,–1). The basis vectors are arranged in the columns of the four-by-four "H₁," then

$$H_1 = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \\ 1 & -1 & 0 & -1 \end{bmatrix} \qquad \text{... (6)}$$

When dealing with four-element one-dimensional signal, the Haar wavelet transform of this signal is a scaling function and three wavelet coefficients. If this Haar wavelet transform is multiplied with Haar basis vectors then the original signal vector will be retrieved. The Haar basis vectors form a matrix that is represented by a variable *H*. If the signal vector is considered as a column vector and is represented by a variable *A* and the scaling function and the wavelet coefficients are considered as a column vector and represented by a variable *W*, then

$$H.W = A \qquad \text{... (7)}$$

$$H^{-1}H.W = H^{-1}A \qquad \text{... (8)}$$

$$W = H^{-1}A \qquad \text{... (9)}$$

$$W = \begin{bmatrix} s \\ w_1 \\ w_2 \\ w_3 \end{bmatrix} \qquad \text{... (10)}$$

$$A = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_1 \end{bmatrix} \qquad \text{... (11)}$$

Where; $H^{-1}$ is the inverse matrix of the *H* matrix (to find the scaling function and the wavelet coefficients, premultiplying by the inverse or $H^{-1}$ in eq. (7)), *s* is the scaling function, $w_{1...3}$ are the wavelet coefficients and $a_{1...4}$ are the signal vector coefficients.

Throughout this work, Fast Haar algorithm is used which will transform the four elements one-dimensional signal into a scaling function and three-wavelet coefficients and this algorithm is called the pyramid algorithm. The components of a signal vector $a_1$, $a_2$, $a_3$ and $a_4$ form the base of the pyramid. Then the average *(x+y)/2*, and the difference *(x-y)/2* are calculated for the pairs $(a_1, a_2)$ and $(a_3, a_4)$. The averages are moved up while the differences are moved down. The resulting averages are used to calculate the averages and differences of the next level, and the average is moved up while the difference is moved down and thus forming the pyramid. The building of the pyramid has transformed the original signal into four coefficients, scaling function that is represented by the top average and the first wavelet coefficient that is represented by the top difference while the second and third wavelet coefficients are represented by the bottom differences. If the Haar basis vectors multiply the resulting coefficients, then the original signal is recaptured.

As one can see, this algorithm is a fast one because it does not contain any multiplication process and it contains only division by 2. This mathematical representation of the data vector is a small representation. This will take small storage area in computer memory and minor consumption through the matching process.

If the signal has *N* samples the multiplication takes $N^2$ steps, while the pyramid does it with only *N* averages and *N* differences (in other words takes 2*N* steps).

Wavelets are a hot topic, but Haar wavelets are cold. Their graphs are made from flat pieces: 1's, 0's, and -1's. Approximation of most signals is very poor. Many flat pieces is needed to represent even sloping line to decent accuracy. This basis will not give compression ratios of 20:1 or 100:1, as desired. Therefore, a better basis is chosen.

The new wavelets are more intricate, but eventually mathematics had to find them. In 1988, at AT&T Bell Laboratories, Ingrid Daubechies found a pulse that starts, stops, and it is perpendicular to all of its dilations and shifts. It is based on four "magic" numbers $d_1$, $d_2$, $d_3$ and $d_4$. Her scaling vector *S* uses them in that order. Her wavelet uses them in the order $W = (d_4, -d_3, d_2, -d_1)$. It is clear why *W* is perpendicular to *S*. The dot product ($S.W = d_1d_4- d_2d_3+ d_3d_2- d_4d_1$) cancels to zero. She also wanted (1, 1, 1, 1) and (1, 2, 3, 4) to have a zero component along *W,* so constant and linear signals can be greatly compressed. Their dot products with *W* must be zero. Then for (1, 1, 1, 1); the dot product = $d_4-d_3+d_2-d_1 = 0$ and for (1, 2, 3, 4); the dot product = $d_4-2d_3+3d_2-4d_1 = 0.$

Those are two equations for the *d's*; two more are needed. The third equation makes the first base tune $(d_4, -d_3, d_2, -d_1, 0, 0)$ perpendicular to the second base tune $(0, 0, d_4, -d_3, d_2, -d_1)$. Their dot product is required to be $d_2d_4+ d_1d_3 = 0$. Then the forth equation is $d_1+d_2+d_3+d_4= 2$ sets the size of the d's; the sum of 2 is convenient. Solving these four equations produces values for the *d's:* $4d_1=1+\sqrt{3}$, $4d_2 =3+\sqrt{3}$, $4d_3=3-\sqrt{3}$ and $4d_4=1-\sqrt{3}$. This gives a much better than Haar, but not the ultimate. Six or eight numbers can be even better, but also more work is required [8, 9].

## 5. The Proposed Algorithm For Daubechies Wavelet Transform

The proposed algorithm for the WT-Daubechies type is constructed from the pyramid algorithm. The bases are chosen to produce best quality for the recovered signal, more security for scrambled signal and manipulate the time for fast calculation. The new Daubechies modification is created in the proposed algorithm to increase the security.

- Modified Daubechies wavelets transform

In this case, *H* is a matrix of 32 by 32, and may be illustrated as two rectangles one over the other as shown below:

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| d1 | d2 | d3 | d4 | d5 | d6 | d7 | d8 | d9 | d10 | d11 | d12 | d13 | d14 | d15 | d16 |
| d32 | -d31 | d30 | -d29 | d28 | -d27 | d26 | -d25 | d24 | -d23 | d22 | -d21 | d20 | -d19 | d18 | -d17 |
| d16 | -d15 | d14 | -d13 | d12 | -d11 | d10 | -d9 | d8 | -d7 | d6 | -d5 | d4 | -d3 | d2 | -d1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| d8 | -d7 | d6 | -d5 | d4 | -d3 | d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 | d3 | d4 | d5 | d6 | d7 | d8 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| d4 | -d3 | d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | d1 | d2 | d3 | d4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 | d3 | d4 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | d1 | d2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d2 | -d1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| d17 | d18 | d19 | d20 | d21 | d22 | d23 | d24 | d25 | d26 | d27 | d28 | d29 | d30 | d31 | d32 |
| d16 | -d15 | d14 | -d13 | d12 | -d11 | d10 | -d9 | d8 | -d7 | d6 | -d5 | d4 | -d3 | d2 | -d1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| d1 | d2 | d3 | d4 | d5 | d6 | d7 | d8 | d9 | d10 | d11 | d12 | d13 | d14 | d15 | d16 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| d8 | -d7 | d6 | -d5 | d4 | -d3 | d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 | d3 | d4 | d5 | d6 | d7 | d8 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| d4 | -d3 | d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | d1 | d2 | d3 | d4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d4 | -d3 | d2 | -d1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 | d3 | d4 |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | d1 | d2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d2 | -d1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d2 | -d1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | d1 | d2 |

**As it can be seen, the basis vectors of the Daubechies wavelet transform are orthogonal, which means that if any vector is multiplied (using inner product) with any other vector the result is zero. To find these 32 elements, 32 equations are required, but some equations are collected in one equation as follows:**

$$\sum_{k=1}^{32} d_k - 2 \qquad \ldots (12)$$

$$\sum_{e=even}^{32} d_e - \sum_{o=odd}^{32} d_o = 0 \qquad \ldots (13)$$

$$\sum_{k-1}^{16} d_k = 2 \qquad \ldots (14)$$

$$\sum_{k=1}^{16} d_k d_{(k+16)} = 0 \qquad \ldots (15)$$

**The equation 16 can be applied as a function to the parameter *j*, where *j* equals 24 and 8 respectively, to produce two equations:**

$$\sum_{k=1}^{8} d_k d_{(k+j)} = 0 \qquad \ldots (16)$$

$$\sum_{k=1}^{8} (-1)^{(k+1)} d_k d_{(25-k)} - 0 \qquad \ldots (17)$$

**The equation 18 can be applied as a function to the parameter *j*, where *j* equals 28, 20, 12, and 4 respectively, to produce four equations:**

$$\sum_{k=1}^{4} d_k d_{(k+j)} = 0 \qquad \ldots (18)$$

**The equation 19 can be applied as a function to the parameter *j*, where *j* equals 29, 21, and 13 respectively, to produce three equations:**

$$\sum_{k=1}^{8} (-1)^{(k+1)} d_k d_{(j-k)} = 0 \qquad \ldots (19)$$

**The equation 20 can be applied as a function to the parameter *j*, where *j* equals 30, 26, 22, 18, 14, 10, 6, and 2 respectively, to produce eight equations:**

$$\sum_{k=1}^{2} d_k d_{(k+j)} = 0 \qquad \ldots (20)$$

**The equation 21 can be applied as a function to the parameter *j*, where *j* equals 31, 27, 23, 19, 15, 11, and 7 respectively, to produce seven equations:**

$$\sum_{k=1}^{2}(-1)^{(k+1)}d_k d_{(j-k)} = 0 \qquad \text{... (21)}$$

$$\sum_{k=1}^{4} d_k = 2 \qquad \text{... (22)}$$

$$\sum_{e=even}^{4} d_e - \sum_{o=odd}^{4} d_o = 0 \qquad \text{... (23)}$$

$$d_4 - 2d_3 + 3d_2 - 4d_1 = 0 \qquad \text{... (24)}$$

## 6. Time Scrambling

In this algorithm, the input signal was first divided into blocks and the samples in each block were permuted. This produces a pronounced loss of intelligibility. However, it also produces a drastic increase in a bandwidth. Further, passing of a scrambling signal through a “real” communication channel and descrambling produces a recovered speech of poor quality. Consequently, generalizations of this algorithm were sought. It was realized that dividing each block into smaller segments of more than one sample, normally around 10 to 30 msec long (which corresponds to 80 to 240 samples at 8 KHz sampling rate), and permuting these segments was a bandwidth preserving operation. It was found that these algorithms were robust in the case of a bad channel (this is illustrated in Figure 2).

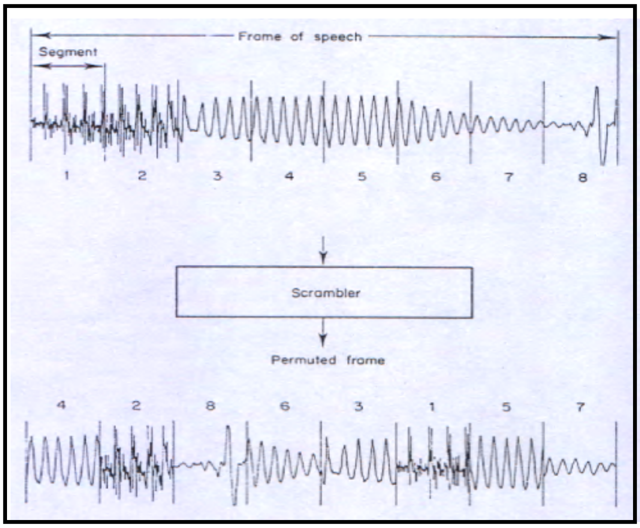

**Figure 2: Speech frame permutation.**

Although the choice of the type of time scrambler part to be used limits the range of rearrangement patterns available, of the types of time scrambler part all require some form of selection process to choose a set of 'good' patterns. For any type of time scrambler part, there are a large number of rearrangement patterns that, if used, result in a transmitted signal that is not sufficiently different from the original signal. If one of these is used then an interceptor can, simply by listening very carefully to the scrambled signal, understand (or at least guess accurately) some of the original message. This is a consequence of a phenomenon known as 'residual intelligibility'. It is sometimes very difficult to predict the level of residual intelligibility of a particular rearrangement. In fact, it is often necessary to scramble a message using the given arrangement then subject the scrambled signal to 'listener tests' in order to see how much of the message is intelligible. There are many different methods for generating usable permutations. Devising and implementing a method for a given system is usually a major part of developed program for a time scrambler part [7,10].

## 7. The Proposed Key Management

Contemporary digital encryption schemes are based on two components; an algorithm that defines the general scheme to be used for protection of the data and a key that makes each instance of the encryption process unique. A major issue in the design of encryption products is how these keys are distributed to each authorized party. This issue is known as key management [11].

There are three keys algorithms that are used: one for time domain, the other for frequency domain, and the third for DWT coefficients. Every key is computed manually according to the following MATLAB program.

```
function [Session_Key] = key_generation (Key_Length)
k = floor(Key_Length/8);n = 0;
for i=1:k
   for j=1:8
      n = n+1; x(i,j) = n;
   end
end
kk = k+1; N = Key_Length-8*k;
if N~=0
   for s=1:N
      n = n+1;y(s) = n;
   end
   Session_Key = [x(:,2)' x(:,5)' x(:,8)' x(:,4)' x(:,1)' x(:,7)' x(:,3)' x(:,6)' y];
else
   Session_Key = [x(:,2)' x(:,5)' x(:,8)' x(:,4)' x(:,1)' x(:,7)' x(:,3)' x(:,6)'];
end
```

## 8. Simulation Results:

The tested speech contains two sentences; Arabic sentence and English sentence. The properties of these waves are shown in table 1. The error rate of the recovered signal equals 10 % and the compression rate of the transmitted signal equals 50 %.

Table 1. The properties of the waves.

| Signal Name | Arabic sentence | English sentence |
|---|---|---|
| Signal Size(byte) | 17600 | 14300 |
| Signal Length (sec) | 2 | 1 |
| Signal Format | PCM | PCM |
| | 6 KHz | 8 KHz |
| | 16 bit | 8 bit |
| | Mono | Mono |

Figures 3 and 4 show; the original signal, the scrambled signal, the descrambled signal, the correlation between the original signal and the descrambled signal, and the correlation between the original signal and the scrambled signal for two different signals.

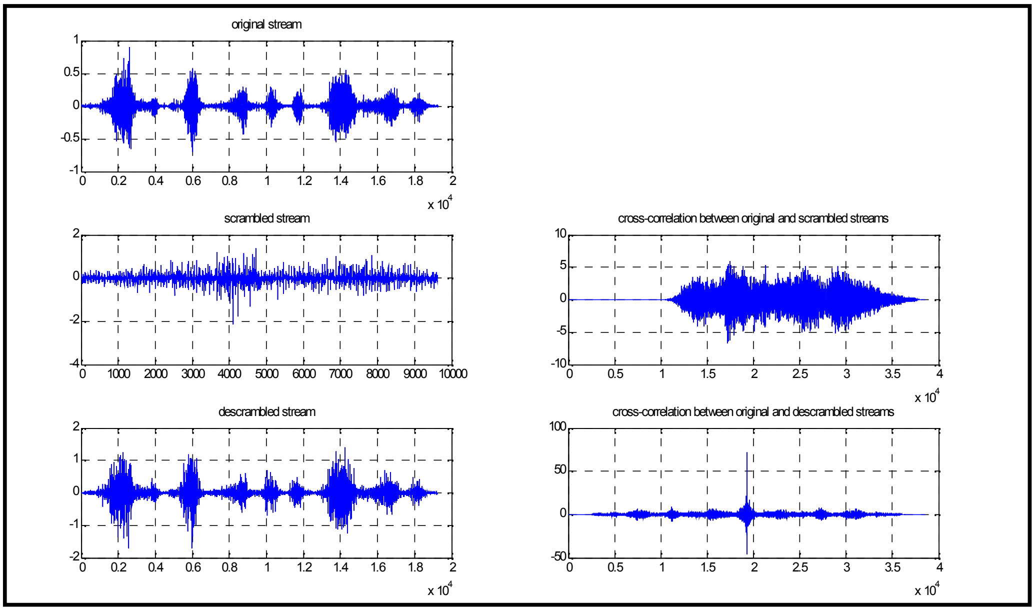

Figure 3: Arabic sentence signal has 16 bits.

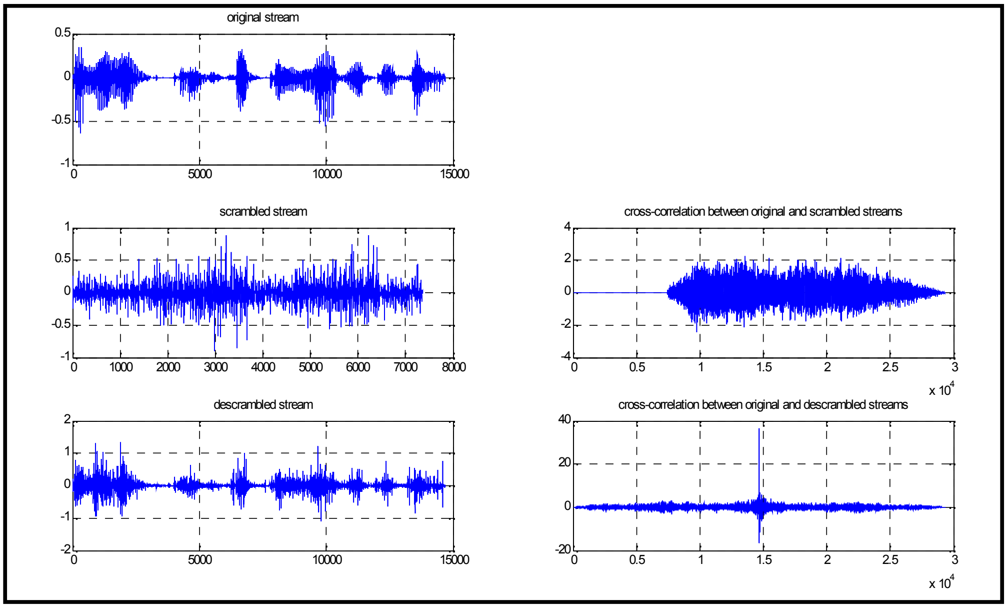

**Figure 4: English sentence signal has 8 bits.**

## 9. Conclusions

1. There are many of impregnabilities that prohibits descrambling, which are the different algorithms for computing keys for time domain, frequency domain, and for DWT coefficients (if the interceptor knows this).
2. The interceptor do not knowing the following factors are the used depth value in the DWT, the type of the transform domain (including the used bases values in this transform), the final form of the speech signal that has half the size of the original speech signal, and the scrambling of the transmitted signal.
3. So scrambling process is stronger compared with most of the scrambling systems that have only two impregnabilities (the used transform domain type and the keys of time domain and frequency domain).
4. From the graphs, the correlation between the original and the scrambled signals leads to good scrambling process. In addition, the correlation between the original and the descrambled signals leads to good descrambling process.
5. The Daubechies modification has a small numbers and its equations are contained on averages and differences only, so it takes a very small size from the memory. Neither uses integration (differentiation) nor uses the complex numbers mathematics operations (i.e. DCT, FFT, Walsh-Hadamard Transform …etc).
6. The bases in the Daubechies modification can be changes as wish. Consequently, it has more secrecy about the transform domain.

7. The present work is successive in most communication applications, data security, and data storing in memory, according to above examples with different frequencies.
8. The residual intelligibility in time domain only is not sufficient to transmit the speech signal, because an enough degree of residual intelligibility to deduce the required information.
9. The residual intelligibility in frequency domain only is sufficient to transmit the speech signal, because the proposal system about the transform domain produces a low residual intelligibility.
10. By using the DWT (and the combined dimensional scrambling), the speech signal becomes sufficient to transmit, and the residual intelligibility is very low.
11. From the compression ratio, there are many advantages; much higher degree of security, lower bit rate, and taking small size (for memory storage and transmission line).
12. DWT technique causes little loss in quality. Nevertheless, they are negligible in terms of cost as compared with the advantages in storage space saving, smaller bandwidth requirements, lower power consumption and small size producing.
13. The quality of the recovered speech depends on the method used in the frequency domain and on the DWT. More clearly, it depends on the type of the transform which is transformed the speech from the time domain to the frequency domain. This transform gives a best representation to the signal. Therefore, the quality of the recovered speech is increased.

## 10. REFERENCES:

1) Lim R. L., **"Speech Compression using the Discrete Wavelet Transform,"** University of De La Salle, Manila, Philippine, -----.

2) Vijay K.M. and Douglas B.W., **"Digital Signal Processing, Hand Book,"** CRC Press LLC, Georgia, 1999.

3) Haykin S., **"Communication Systems,"** John Wiley and sons Inc., New York, 2001.

4) Jayant N.S., **"Analog Scramblers for Speech Privacy,"** In Computers and Security 1, New York, 1982.

5) Matsunaga A., Koga K. and Ohkawa M., **"An Analog Speech Scrambling System Using the FFT Technique with High-Level Security"**, IEEE, Transaction on Communications, Vol.7, May 1989.

6) Weeks M., **"Digital Signal Processing Using MATLAB and Wavelets,"** Georgia State University, infinity science press LCC, Hingham, Massachusetts, 2007.

7) Bopardikar A.S., **"Speech Encryption Using Wavelet Packets,"** M.Sc. thesis, Indian Institute of Science, 1995.

8) Strang G., **"Wavelets,"** American Scientist, Vol. 82, pp. 250-255, May/June 1994.

9) Ingrid Daubechies, "**Ten Lectures on Wavelets,"** Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 1992.

10) Beker H.J. and Piper F.C., **"Secure Speech Communications,"** London, Academic Press, 1985.

11) Wenbo M., **"Modern Cryptography: Theory and Practice,"** Prentice Hall PTR, New Jersey, 2003.

# ضغط وتشفير الاشارة الرقمية بأستعمال التحويل المويجي

م.م.عماد حمود سلمان

جامعة ديالى

## المستخص:

بالتطور السريع لتكنلوجيا الوسائط المتعددة والشبكات, اصبحت امنية هذه التكنلوجيا ضرورية اكثر فأكثر, هذه البيانات الوسائط المتعددة ترسل عبر عدة طرق. ان الامنية ذات المصداقية في الخزن والارسال لبيانات الكلام والصوروالصور المتحركة (الفديو) الرقمية نحتاج اليها في عدة تطبيقات مثل انظمة الصور الطبية وقواعد بيانات الصور العسكرية خصوصا وفي المؤتمرات التلفازية. في الاونة الاخيرة ظهرت بعض الاجهزة الالكترونية مثل الهاتف النقال الذي يملك وظيفة تبادل البيانات الكلامية والموسيقيةو الصورو المقاطع المتحركة عبر خدمة رسائل الوسائط المتعددة عبر الشبكة اللاسلكية, كل ذلك يتطلب امنية للوسائط المتعددة.

الطريقة المقترحة ذات امنية عالية الدرجة وذلك بحساب التحويل المويجي المنفصل ومعكوسه بواسطة الحوارزمية الهرمية. حيث انها تسمح ببناء قواعد لذلك التحويل, هذا يقود لمزيد من السرية حول تلك القواعد في المنظومة. ايضا هذه الخوارزمية تملك معالجة مساوية لضعف عدد عينات الاشارة الاصلية فقط وليس لمربع عدد عينات الاشارة الاصلية كما في اغلب التحويلات. يتضمن البحث التمثيل الرياضي لتعديل التحويل المويجي المنفصل ومعكوسه متبوعا بمرحلة ثانية من التحويل المويجي المنفصل لكبس الاشارة. وعليه فأن المنظومة تشفر الاشارة بواسطة البعدين (الوقت والتردد) ثم تضغط وتشفر مرة اخرى. أن المفاتيح المستخدمة خلال هذا البحث ولدت يدويا: الاول لبعد الوقت والثاني لبعد التردد والاخير لعينات الاشارة المضغوطة.

واخيرا, كل الطرق السابقة اعطت اشارة مضغوطة ومشفرة وغير مفهومة واشارة مفتوحة الضغط والتشفير ومفهومة اعتمادا على النتائج العددية والشكلية, وأن نسبة الضغط كانت مساوية الى 50٪ ونسبة الخطأ مقاربة الى 10٪.