GENETIC OPTIMIZATION OF WAVELET PACKET COMPRESSION FOR SPEECH SIGNAL

Leonardo CERNĂIANU, Eugeniu OANCEA, Alexandru ŞERBĂNESCU

Academia Tehnică Militară, Blvd. George Coşbuc 81-83, Sect. 5, Bucureşti

The compression of speech signals using Wavelet Packet (WP) consider fixed values of threshold for elimination of small coefficients from resulting subspaces, but none of them take into account the speech specific characteristics. This leads to rude distortions of high frequencies of speech signal and small compression degrees. In this paper, we propose the use of genetic algorithms (GA) for finding multiple threshold values (one for each subspace of WP decomposition) to achieve the maximum compression and the maximum quality of the synthesized signal. We transformed the multiobjective optimization algorithm with controversial fitness functions (quality F1 and compression F2) into an simple one by building a self-adaptive function witch take into account the both targets. The final solution will be determined from the Pareto space generated by the F1 and F2.

Key words: Speech signal, multiobjective genetic algorithms, wavelet packet; compression, adaptive.

1. INTRODUCTION

Speech compression has been one of the most studied fields in the past decades. The communication needs increased over time and bandwidth management required more and more attention. The necessity for speed and quality leaded to development of more complex algorithms for speech analysis and compression such as Wavelet Packet (WP) decomposition method.

The typical algorithm, as presented in [1] and [2], for the WP compression makes use of a fixed (single) threshold (T) for elimination of small coefficients from decomposition subspaces. For the particular case of a speech signal, one can observe that vowel information concentrates in high values coefficients and that consonant leads to small ones. By imposing a threshold *T* of a high value, common for all subspaces, the synthesized signal will loose a part from the high frequencies of the input signal (low amplitudes which are characteristic for consonants) as shown in the picture below.

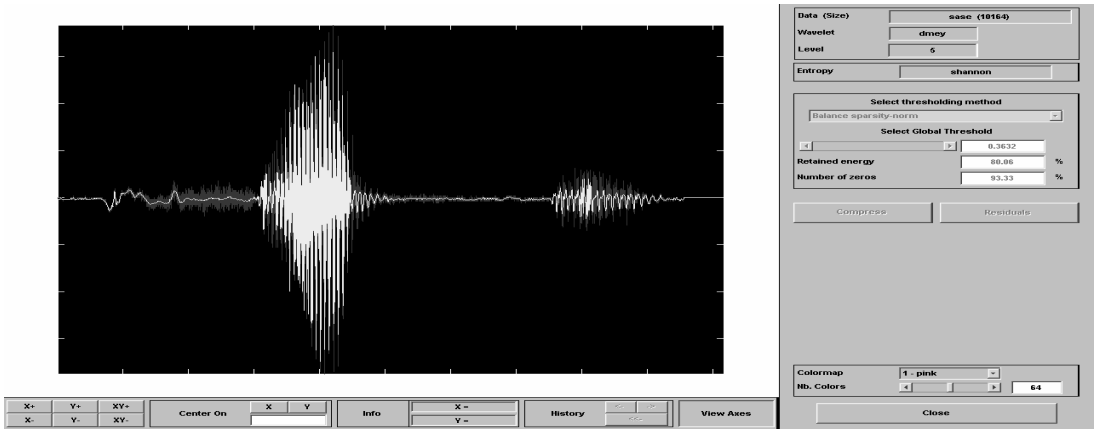

Figure 1. The input signal and the synthesised one from the WP coefficients.

Decreasing the value of *T*, more and more information will be eliminated from the analysed speech signal, but the compression degree will increase. From this rises the question of choosing the value of

threshold. There are different proposed methods of calculating the value of T, but none of them was found by us to provide a good synthesised speech signal and a good compression degree in the same time.

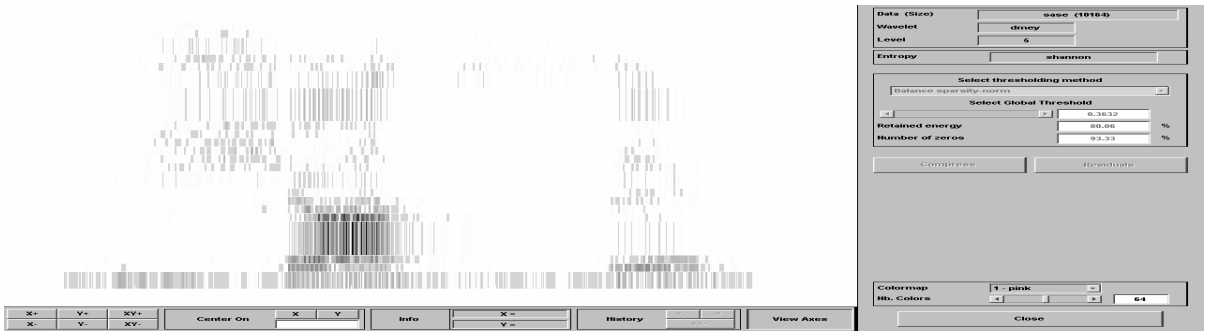

Figure 2. The WP coefficients before the compression.

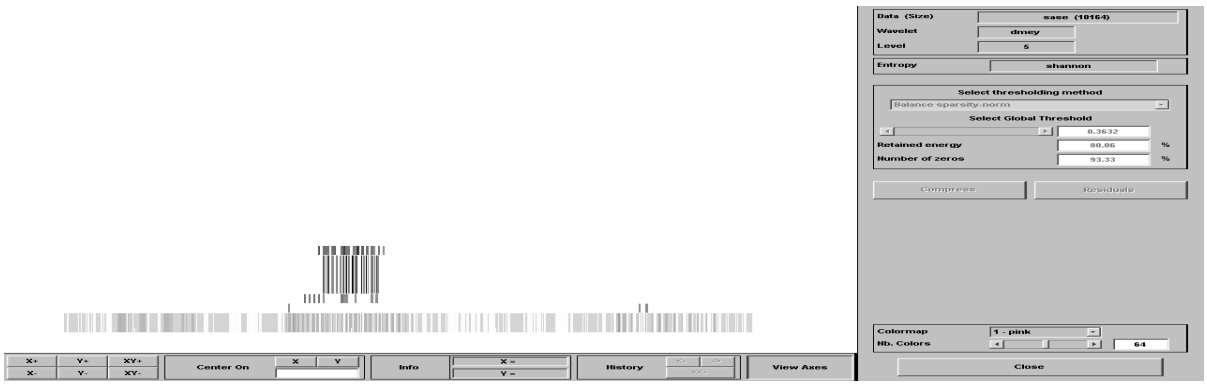

Figure 3. The WP coefficients after compression.

A natural approach for solving this problem is the use of multiple threshold values (T_i), one for each interval. But this leads to another problem – which are the optimum values of T_i able to provide the target we search?

2. AN OPTIMUM PROBLEM

A very good method of finding a solution for an optimum problem is the use of genetic algorithms. Those algorithms can find a local or global minimum of a fitness function [3]. Unfortunately we have two – the compression and the quality.

A multifunction genetic algorithm is hard to implement especially when the fitness functions are antagonist. The common approach is to make use of Pareto front (the Pareto plane is the one generated by (F1, F2) points as shown below and the Pareto front is defined by the multitude of points which are closer to the axes) and to chose the solution we need, but this cannot be done without a carefully chose of the fitness functions.

The algorithm we propose consider the WP decomposition of speech signal [1], find the best basis and try to determine the values of thresholds T_i. To achieve this, we consider the following:

– The chromosome size will be C_s, the same as the number of intervals in the WP best basis decomposition;
– The initial individuals values will be uniform distributed between the minimum and the maximum values of the best basis coefficients;
– The population size P_s will be of 100. This value is big enough to confer a good covering of the coefficients interval and small enough for the algorithm to converge quickly to a optimum solution;
– The maximum number of generations G_s will be 200;

– The elite population size will be calculated as $\frac{P_s}{2}+10$ (for a quick convergence, more then half of the population will be transmitted).

As we said, we have to consider two fitness functions – one for compression and the other for quality. For a given chromosome (T_i) we will define the two functions as following:

a. The quality function (F_1)

$$F_1 = \frac{1}{SNR} \tag{1}$$

where:

$$SNR = -10 * \log 10\left(\frac{E_{imp} - E_r}{E_{imp}}\right)$$

E_{imp} = the desired energy of the synthesized signal, percentage of $E_{CoefInit}$

$E_r =$ the resulting energy of the decomposition coefficients, after thresholding, percentage of $E_{CoefInit}$

$$E_r = \frac{E_{CoefTrh}}{E_{CoefInit}} * 100$$

$E_{CoefTrh} =$ the resulting energy of the decomposition coefficients, after thresholding;

$E_{CoefInit} =$ the energy of the decomposition coefficients, before thresholding.

It is obvious that we are looking for a synthesised signal as close as possible to the input signal. This is the reason for imposing the condition for the desired energy to be of 100% from the input signal. Taking into consideration the principle of energy conservation, the energy of the WP coefficients, before applying the threshold, is the same as the energy of the input signal. This is the motivation for calculating all the energies as percentage from the coefficients energy.

As shown above, we considered as "noise" (distortion) the difference between the imposed signal (the same as the energy of the coefficients) and the resulting (synthesised) one. From this point of view, the quality function (inverse of the SNR) is equivalent with the condition as the distortion to be as small as possible (the difference $E_{imp} - E_r$ to tend to zero).

The relation between the retained energy (the energy of the synthesised signal) and the SNR can be obtained by replacing E_{imp} with 100 in the SNR formula. We can represent the result as in the following picture:

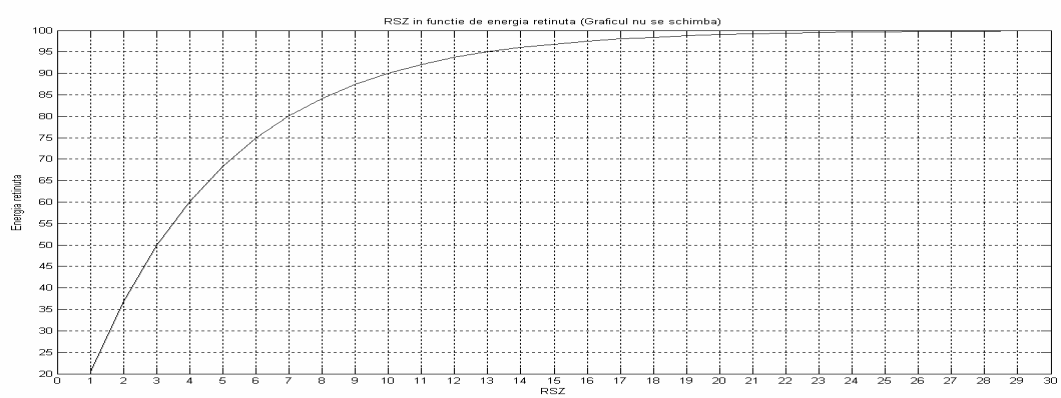

Figure 4. The relationship between the retained energy and SNR.

b. The compression function (F_2)

$$F_2 = \frac{CNZ}{N} \tag{2}$$

where:

$CNZ =$ the number of nonzero decomposition coefficients, after thresholding

$N =$ the number of decomposition coefficients, before thresholding

It is easily seen that as the number of nonzero decomposition coefficients *CNZ* grows, the compression degree will decrease but we will have a better quality of synthesized signal so the two fitness functions F_1 and F_2 cannot reach an optimum value for the same considered T_i. Instead we can try to find an individual T_i which provides as minimum values as possible for the F_1 and F_2 functions. Anyway, we must provide to the optimization algorithm a single fitness function even if we use the Pareto method. We cannot choose F_1 or F_2 because we want an optimum for both so we must build a new fitness function.

Thinking of Pareto plane, the solution which minimizes both fitness functions is the one closest to origin (0, 0). We can calculate the Euclidian distance to origin as following:

$$rF = \sqrt{F_1^2 + F_2^2} \tag{3}$$

But F_1 and F_2 are both positive so we can redefine the *rF* function as:

$$rF = F_1 + F_2 \tag{4}$$

Choosing the fitness like this we will find a sum of points spread in the Pareto plane which converge to a minimum point. But this function does not provide a quick convergence and a lot of points are not useful. At every step of the algorithm we must store the F_1 and F_2 values in order to build the Pareto plane, but those values can be used also to change the formula in such a way that the compression degree is bigger at every step of the algorithm. To achieve this, we change the fitness function at every step by imposing the compression degree to be bigger than the maximum obtained until that moment:

$$rF = F_1 + \left|\min(F_2) - F_2\right| \tag{5}$$

With this fitness function, we obtain the Pareto plane shown in the following picture. For the analyzed word "sase" we considered the sampling frequency of F_s=11025 Hz, a B_s=8 bits/sample (256 levels of amplitude) so a bit rate of 88200 bit/s. For the WP decomposition we used the hard threshold method for eliminating the coefficients, the digital Meyer function as mother function and the Shannon entropy as the cost function for determining the best basis. This must be considered just an example. The input signal can have any sampling frequency and can be considered any number of bits for coding the speech sample without any change in the algorithm structure. The same for the WP decomposition algorithm – any desired parameters can be chosen, but those were found by us to be the best for a speech signal compression application. In general, the bit rate of the compressed signal can be calculated as $B_S * F_s * F_2$.

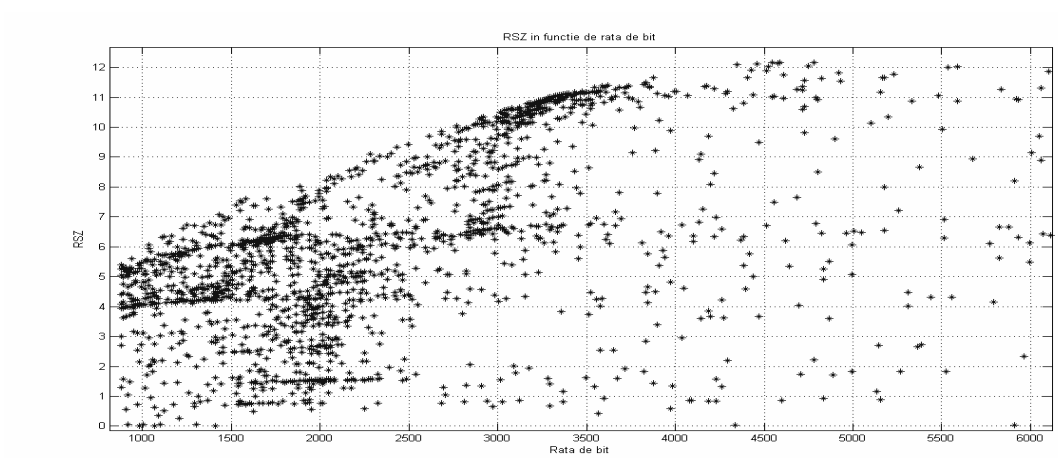

Figure 5. Pareto plane – SNR as function of the bit rate.

As one can observe in the above picture, the fitness function provides solutions that confer a good quality - there are solutions that confer a retained energy above 90% (SNR above 10) -, but the compression (bit rare) is distributed along the axis. If we want a bit rate under 3000 (for example, 2400 is the standard for LPC10 [5]), one can observe that there are very few points in the Pareto plane to choose from (to have also a good quality). This is because the fitness function converges slowly to bigger compressions while trying to maintain the quality. Doing so, a lot of individuals are lost (the solutions they provide are not of interest). Instead, it is possible as we to want more solutions at higher compression rates and less at smaller

compression degrees or vice versa. It is important to remember that a bigger compression leads to a worse quality of the synthesized signal.

From those observations we decided that it will be useful as the solutions the algorithm generates to converge to a bigger compression or to a bigger quality depending of the input signal characteristics (if the signal has a consonant or a vocalic nature). This can be realized by transforming the formula given in (5) as following:

$$rF = w_1 * F_1 + w_2 * \left| \min\left(F_2\right) - F_2 \right| \tag{6}$$

where w_1 and w_2 are some weights chosen to provide more compression or more quality.

By choosing a bigger w_2, we concentrate the solutions at bigger compression rates (smaller bit rates) by keeping a reasonable quality, but with the raise of w_2 after a given value, the quality begins to decrease. It must be kept in mind that the role of w_2 is just to concentrate the population at higher compression rates so it cannot be too big because it will decrease the quality. This leads as to idea that, if we use consonant speech segments, we can impose a bigger compression. In the case of vocalic segments we need quality as the synthesized signal to be as close as natural speech as possible.

In the following figure are presented two cases – the first one for w_2 having the value of 200 and the second for w_2 of 20:

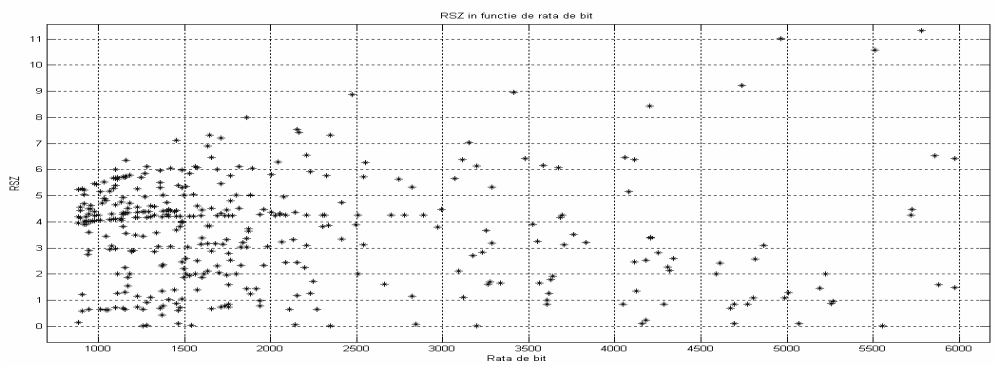

Figure 6. Pareto plane for w_2 of 200 (SNR in function of bit rate).

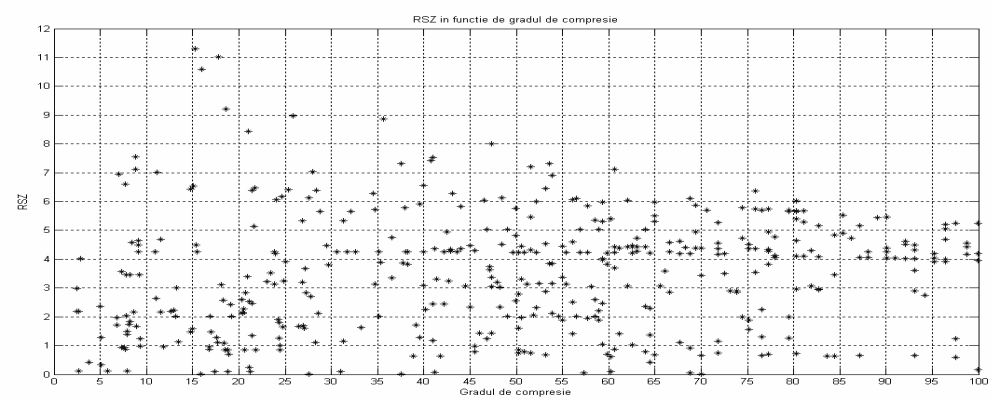

Figure 7. Pareto plane for w_2 of 200 (SNR in function of the compression degree).

In the first case, the solutions of interest concentrate around a bit rate of 1000 - 2500, but at a SNR of maximum 7 – 9 which means that the synthesised signal will retain 80 - 87% of the input energy. In the second case, the solutions of interest concentrate around a bit rate of 1000 – 3500 but at 3500 we have a SNR of even 10 which means that the synthesised signal will retain 90% of the input energy. Depending of the application, a retained energy (quality) of 80% for a speech signal (especially for a vocalic segment) can be insufficient. If we want to improve the quality, it is sufficient to increase the w_1 value. In the following example we have $w_1 = 10$ and $w_2 = 20$.

3. COMPRESSION OR QUALITY

As we said in the beginning of this paper, in the case of WP speech compression, the information regarding the superior part of the spectrum of signal is eliminated in order to reduce the quantity of data. This makes the consonant part of the synthesized signal to badly reproduce the initial signal. Taking into account the effect of integration of the human hearing and the fact that most of the information transmitted is carried by vowels, the lost of a part from the consonant information is not a problem. This is not the case of the vowels. From this point of view, to achieve a bigger compression while keeping a good quality, it is a good idea to implement a segmentation of the speech signal before compression. This segmentation can also be implemented using time frequency techniques.

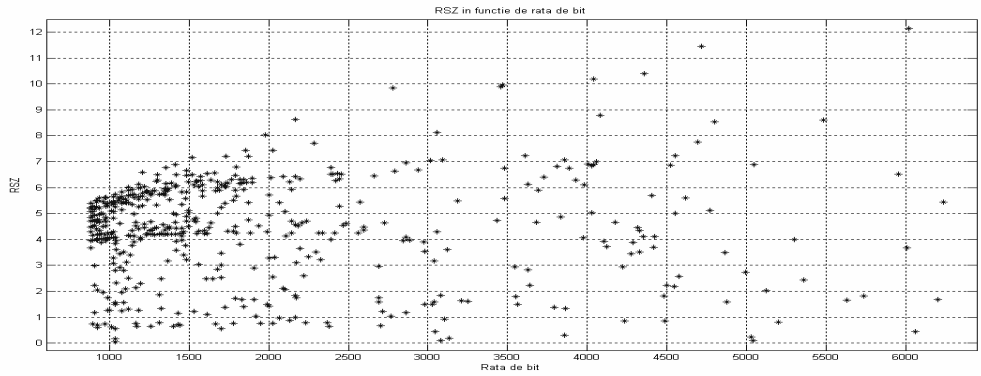

Figure 8. Pareto plane for w_2 of 20 (SNR in function of bit rate).

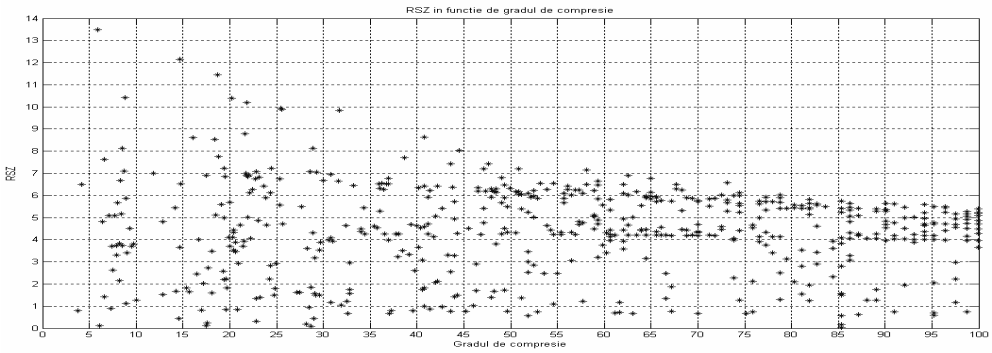

Figure 9. Pareto plane for w_2 of 20 (SNR in function of the compression degree).

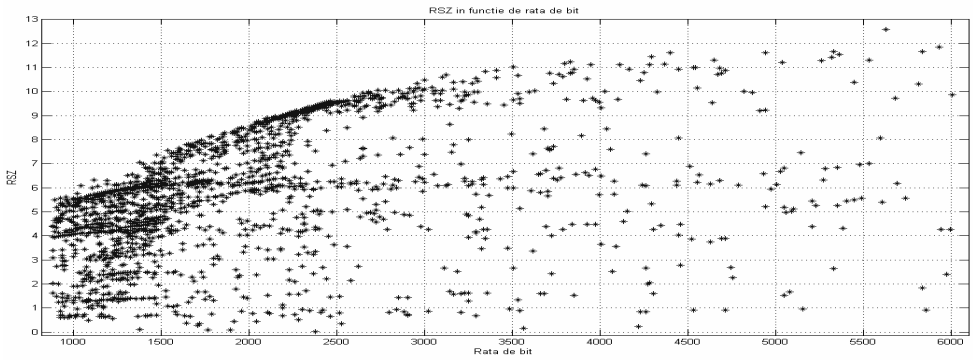

Figure 10. Pareto plane for w_1 of 10 and w_2 of 20 (SNR in function of bit rate).

For the presented compression algorithm, we choose a special segmentation method – we split the speech signal in consonant and vocalic parts instead of vowels and consonant elements. The main difference

is that the vocalic parts can contain vowels and consonants, but the energy of vowels which is diffused in the transition areas raise the energy of the consonant and therefore it can not be ignored anymore. For example, the sequence "dul" will be split in two parts – "d" and "ul". The second part contain one consonant ("l"), but the vowel "u" transfer to the consonant "l" a part from the characteristics of a vowel – energy, a more complex spectrum containing also low frequencies, etc. In fact, in this case, the consonant identifies itself with a transition from the vowel to the silence.

The segmentation method that we propose decomposes the speech signal using multi resolution analysis method (MRA) [4] on level 3 and calculates the energy of the details at each level. We notate those energies with E_{d1} for first, E_{d2} for second and E_{d3} for third level of decomposition.

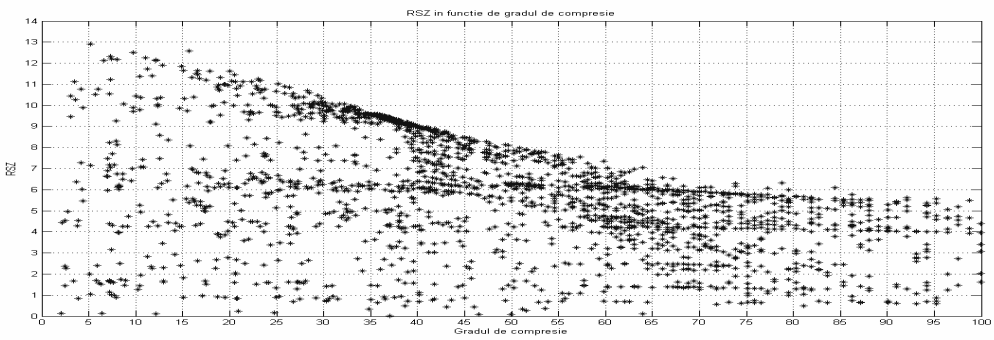

Figure 11. Pareto plane for w_1 of 10 and w_2 of 20 (SNR in function of the compression degree).

The choice to decompose the signal on level 3 came from practical experience – continuing the decomposition we will obtain energy of detail at level N (notated E_{dN}) smaller than E_{d1} even if the segment is vocalic or consonant. More, if we consider that the spectrum of the approximation signal is split in half at each decomposition step and a maximum frequency of the input signal of for example 5000 Hz (a sampling frequency of 11025 Hz), we observe that the spectrum for the approximation signal is limited at 2500 Hz for the fist level, at 1250 Hz for the second level and at 625 Hz for third level. At the next level, the approximation signal will have a spectrum limited at 312 Hz and the detail signal will be limited between 312 Hz and 625 Hz. As you can see, the spectrum of the detail at level 4 is characteristic for a vowel and the one of approximation contain (or must contain in normal circumstances) just noise.

The spectrum of a consonant is characterised by low energy in contradiction with the one of a vowel. In those conditions we can decide that a segment is consonant if we have the relationships:

$$\begin{cases} E_{d1} - E_{d2} > 0 \\ E_{d1} - E_{d3} > 0 \end{cases} \tag{7}$$

We must specify that the level of decomposition depends on the sampling frequency. If the sampling frequency is bigger than 11025Hz for example, it might be necessary to increase the level of decomposition and the number of inequalities.

In the following figures we present two cases – first one, the Romanian word "şase" and, second, the sequence "dul".

Using this method of segmentation, we can utilize in the formula (6) a bigger w_2 and a smaller w_1 if the sequence is considered consonant, but a smaller w_2 and a bigger w_1 if the sequence is vocalic.

If we want to compress a speech signal, first we split it using the presented method then, depending of its character; we build the fitness function using different values for w_1 and w_2. To choose the values of those parameters, we will take into account the following:

- The w_2 will be direct proportional with the divisions $D_2 = \frac{E_{d1}}{E_{d2}}$, $D_3 = \frac{E_{d1}}{E_{d3}}$ and the elite population size;
- The parameters w_1 and w_2 are inverse proportional (in the way that if we increase one, we must decrease the other one).

4. CONCLUSIONS

In this paper, we have presented a technique for optimum Wavelet Packet compression of speech signal and gave a solution for the problem of choosing the threshold values. In the same time we have proved

that the genetic algorithm can be used with success in solving optimization problems and showed that in combination with classical algorithms provide good and, most of the time better, solutions at common issues.

We have shown that instead to consider two objective/fitness functions – one for compression and the other for quality we can use a monocriterial optimization using the formula given in (5) where w_1 and w_2 are weights chosen to provide more compression or more quality. Using the method of segmentation proposed in the paper, we could utilize in the formula (6) a bigger w_2 and a smaller w_1 if the sequence is considered consonant, but a smaller w_2 and a bigger w_1 if the sequence is vocalic.

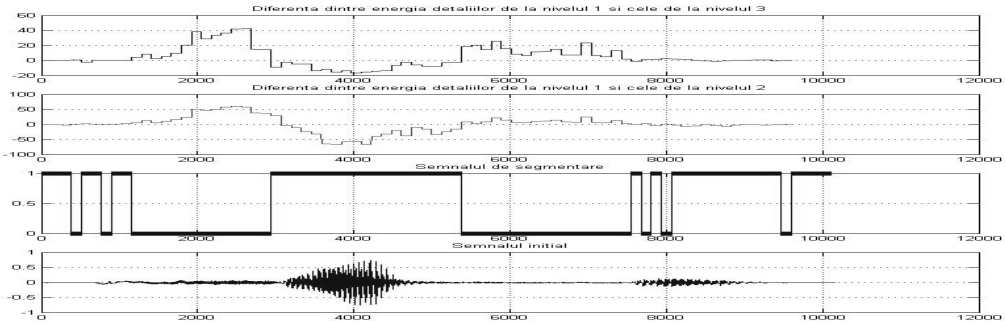

Figure 12. The segmentation of the Romanian word "şase". From up to down: E_{d1} - E_{d3}; E_{d1} - E_{d2}; the segmentation signal; the input signal.

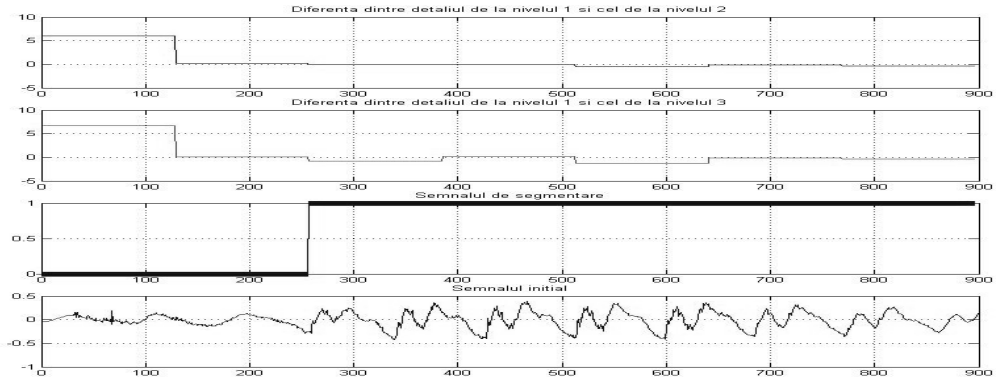

Figure 13. The segmentation of the sequence "dul". From up to down: E_{d1} - E_{d2}; E_{d1} - E_{d3}; the segmentation signal; the input signal.

The algorithm we propose offer good bit rates (compared for example with LPC10) and good quality of the synthesised speech signal. The method can be implemented using quick calculation algorithms in both the segmentation and WP decomposition parts. Taking into consideration that the GA can be forced to converge to a solution (as shown above) and that it accept stopping conditions, we ca say that the shown method can be implemented in real time even if we didn't have the opportunity to do it.

An improvement of the proposed algorithm can be obtained if we take into consideration the masking effect, but this will be the object of the next stage of the algorithm development. Not considering the method as being fully optimized, we tried to show that the combination of the wavelet packets and genetic algorithm as new techniques of signal processing leads to results that cannot be obtained with classical algorithms.

REFERENCES

1. MALLAT S., *A Wavelet Tour of Signal Processing*, Second Edition, Academic Press, 1999.
2. ISAR A., NAFORNITA I., *Reprezentări timp- frecvenţă*, Timişoara, Editura Politehnică, 1998.
3. MAN K.F., TANG K.S., KWONG S., *Genetic Algorithms*, Springer-Verlang, London, 1999.
4. AKANSU N, HADDAD R., *Multiresolution signal decomposition*, Second Edition, Academic Press, 1992.
5. HUANG X., ACERO A., HON H., *Spoken Language Processing – A Guide to Theory, Algorithm, and System Development*, Prentice Hall Press, 2001.