Evolutionary Algorithm for Speech Segmentation

D. H. Milone , J. J. Merelo and H. L. Rufiner

Abstract - Speech segmentation is one of the problems in speech processing area. The main techniques that attempt to solve it are manual segmentation and hidden Markov models alignment. In this work a new technique based on an evolutionary algorithm that permits to segment the speech without previous training process is presented.

Keywords—**Speech segmentation, evolutionary algorithm, speech distance measurements, HMM alternatives.**

I. Introduction

Speech segmentation (SS) consists of the division of an emission in different chunks according to some criterion. SS is commonly used for separating speech into phonemes, syllables or superior level units, such as words [1], [2].

In the simplest case, the SS problem deals with finding the accurate limits that define each segment or phonetic unit. Each segment presents two limits or markers that measure the times, from the beginning of the emission, in which the beginning and the end of the segment in question are found. An emission can have many segments and thus the correct location of all its limits can be a complex problem, even more so if all the speech variations associated are considered, as generally happens in speech related problems.

Several techniques have been used for SS. *Manual segmentation* was the first: an expert linguist generates the segmentation based on spectrograms, energy curves, intonation and other techniques used in speech analysis. This technique possesses the advantage that the linguist experience assures a very good result in the segmentation. However, the costs in time and resources that this manual process carries are the highest and makes it only applicable to very specialized studies. The second technique applicable to segmentation comes from automatic speech recognizers. In automatic speech recognition, the *hidden Markov models* (HMM) technique currently gives the best results [3]. Upon applying HMM to automatic speech recognition, there exists an implicit segmentation process (model alignment) and, with some modifications to reduce the computational cost of a complete recognition, it is possible to use them stand-alone for SS [4]. However, the classical methods based on HMM alignment requires the full transcription of the speech input, in other words, a full speech-recognition process is needed.

There are also other alternative methods performing SS, not necessarily limited to speech processing techniques but rather methods of generalized application. For instance, neural networks [5], [6], statistical modeling [7], and parametric filtering [8]. In any case, automatic SS problem remains unsolved and even less in real time applications.

An important application is the segmentation of speech databases, in this case the corresponding transcription was generally given and fast algorithms are not necessary. In the other hand, if the segmentation is required as part of a process to recognize speech, obviously transcription was not given and it is required to work in real time. Therefore the characteristics of the suitable algorithm depend on the final application. Different evolutionary computation (EC) methods have offered a solution to many problems in the last decade, mainly search and optimization [9]. They have been applied with success to image segmentation [10], for example. In the particular case of this work, the objective is an EC algorithm to solve the automatic SS problem.

II. Speech Segmentation through EC

In this section, the definition of the problem is formalized and the design of the evolutionary algorithm for SS is described. Initially speech signals, its parameterization and the segmentation process are presented. Then the evolutionary algorithm is defined: representation of the individuals, fitness function, selection method and reproduction and its operators. Finally, some implementation details are mentioned.

A. Speech signal and its parameterization

Let $s(t)$ be the continuous speech signal for the real time variable t. After a sampling process with sampling period $T_S = 1/f_S$, the discrete time speech signal is represented as $s(nT_S)$ or simply $s(n)$, for the discrete time natural variable $0 < n \leq N_S$.

Diego H. Milone and H. L. Rufiner are with Cybernetics Laboratory, Faculty of Engineering from National University of Entre Ros (UNER), Argentina. His work is supported by PID 6036-1/UNER.

Juan J. Merelo is with Department of Computers Architecture and Technology from Granada University, Spain. Their work is supported by FEDER I+D 1FD97-0439, CICYT TIC 1999-0550, DREAM IST-1999-12679 and INTAS 97-30950.

If $\omega(n)$ is an analysis window defined in $0 < n \leq N_\omega$, it is said to have a *width* $N_\omega T_S$. If this window is displaced in regular intervals of time $N_D T_S$ then we can define:

$$\omega_i(n) = \begin{cases} \omega(n - iN_D) & \text{if} \quad 0 < n - iN_D \leq N_\omega \\ 0 \quad \text{otherwise,} & \end{cases}$$

and calling it *step* of the analysis window at the time $N_D T_S$. Given the previous definitions, the independent variable i remains bounded according to $0 < i \leq N_I$ being $N_I = (N_S - N_\omega)/N_D + 1$.

Given the transformation Ψ, the speech signal parameterization process is accomplished according to $c_i(k) = \Psi\left(\omega_i(n)s(n)\right)$, being $0 < n \leq N_S$ the independent time variable, $0 < k \leq N_K$ the independent variable in the transformed domain and $0 < i \leq N_I$ the displacement of the analysis window.

B. Segmentation

The segmentation process consists of the speech signal markup according to given distinctive characteristics. Considering the speech signal parameterization according to the previous description, the segmentation yields as result a set of segments $\Phi = \{E_m\}$ with $0 < m \leq N_\Phi$ where each segment E_m contains features vectors $c_i(k)$ with given degree of membership. On this general definition, two restrictions will be made. The first is to consider the totally exclusive segmentation, where each feature vector can belong to only a segment. This permit to describe the membership without a membership function associated with each feature vector. The second restriction is that the temporary order according to the one which the feature vectors appear in the segments can not be inverted. The two restrictions can be expressed together through the following equation:

$$c_{i_1}(k) \in E_{j_1} \wedge c_{i_2}(k) \in E_{j_2} \quad \Leftrightarrow \quad i_1 < i_2 \,\forall\, j_1 < j_2$$

Given these restrictions then segmentation can be represented through the markers vector of the first element of each segment $\boldsymbol{\phi} = [M_j]$ with $0 < j \leq N_\phi = N_\Phi + 1$, since the initial and final markers are included and furthermore $1 \leq M_1 < M_2 < \ldots < M_{N_\phi} \leq N_I + 1$.

As it will be seen it is convenient to leave open the possibility that the first marker will be greater than 1 and the last less than $N_I + 1$. Strictly the matrix $c_i(k)$ is not defined in $i = N_I + 1$ but the marker for the formalization of the fitness function will be valid.

C. Representation of individuals

The first aspect to solve in EC algorithm design is the problem of codification in a finite alphabet. Traditionally binary strings (pure GA) have been used, but currently more flexible data structures are being used [11], [12].

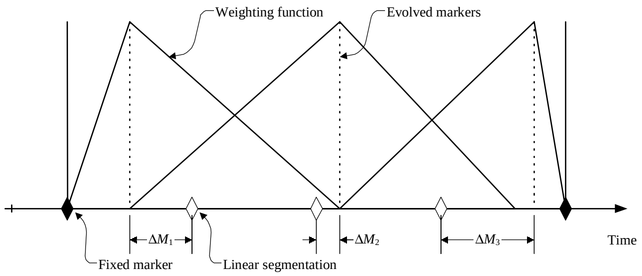

Fig. 1. A schematic example of evolved delta-markers from linear segmentation of speech signal. In addition, the weighting function $\alpha(\cdot)$ is shown for evolved delta-markers.

The genetic material of each individual will be a set of segmentation markers. This codification will take as starting point the linear segmentation of speech utterance. The working hypothesis is that the number of segments N_ϕ (but not the transcription) of speech signal is known in advance; later on a method to eliminate this restriction will be discussed.

The linear partition consists of assigning the markers of each segment according to $M_j = M_1 + (M_{N_\phi} - M_1)(j - 1)/(N_\phi - 1)$ for $1 < j < N_\phi$, where the initial and final markers need not be 1 and N_I respectively. In fact, a detector of beginning and ending of the utterance was implemented based on the by-windows energy analysis of $s(n)$. This permits an important reduction of the search space for segmentation.

From this linear segmentation the displacement of the markers can be defined as $\boldsymbol{\Delta\phi} = [\Delta M_j] \quad$ with $\quad 1 < j \leq N_\phi - 1$, which will be used as the data structure to evolve (see Figure 1). The displacement vector $\boldsymbol{\Delta\phi}$ does not include displacements for the first and last markers because they are not important in the segmentation process and remain fixed. The displacements for the markers ΔM_j are integer numbers in a range determined by the maximum possible lengths for the segments. In the case of phoneme segmentation, a displacement range of 50 ms is sufficient. However, for syllable segmentation, the range can be up to 200 ms. If the segmentation had included the silences at the beginning and at the end of the emission, it would been have necessary to increase range up to 1 second.

So, the genetic material of each individual will be a vector of integers (with bounded and clean-defined range). Each integer will represent the displacement from the linear segmentation markers, excluding the first and last.

The method to obtain the markers from the information codified in the genetic material of each individual is straightforward.

Some other problems related to the evolution process itself are also solved. Since a codification of problem solutions and not the solutions themselves is evolved, it is possible that during evolution the genetic material result in non-valid phenotypes: individual not compatible with the life or incoherent solutions. In this particular problem and given the elected codification, the solutions can be invalid in two cases: a) when decoding the markers their natural order is not respected, that is to say (1) is violated and overlaps are produced and b) when one or more markers are outside of time limits of the speech utterance. This possibility exists independently of the first case since the initial and final markers are not evolved.

The problem can be solved in many ways [11]. For example, it is possible to choose a codification that does not permit these genetic mistakes after the application of the different operators. As well, genetic operators might not permit the generation of wrong chromosomes starting from valid chromosomes.

A simpler technique that does not imply an important modification in the original idea of EC is to check and repair the gene values when they are written. In order to check overlapping, the following inequation should be met $M_{j_1} < M_{j_2} \, \forall \, j_1 < j_2$, with $1 < j_1, j_2 < N_\phi$.

Checking is completed verifying that no marker is found outside speech utterance limits, determined by the initial and final markers. All can be summarized widening the ranges in the prior expression to: $1 \leq j_1, j_2 \leq N_\phi$.

D. Fitness Function

The segment characteristic vector is defined as:

$$\varphi_j(k) = \frac{1}{A_j(\cdot)} \sum_{i=M_j}^{M_{j+1}-1} \alpha(\cdot) c_i(k)$$

where

$$A_j(\cdot) = \sum_{i=M_j}^{M_{j+1}-1} \alpha(\cdot), \quad \text{with} \quad 0 < j < N_\phi - 1$$

Weighting function $\alpha(\cdot)$ is devised to assign different weights to feature vectors according to their distance of the segment limit. Weighting function can be defined, for instance as $\alpha(d, N) = e^{-d/N}$ or as a linear relationship $\alpha(d, N) = 1 - d/N$, being d the distance to the marker and N the total number of vectors to weight (see Figure 1). Supposing the linear relationship is adopted and that d goes from 1 until N, it can be proved that $A(d, N) = \sum_d 1 - d/N = (N + 1)/2$.

To distinguish between a segment characteristic vector weighted as previous or next to a marker, the superscripts '−' and '+' will be used, respectively. Equations below of the characteristic vectors of a segment are presented according to their marker relative position:

$$\varphi_j^-(k) = \frac{\sum_{i=M_j}^{M_{j+1}-1} \alpha(M_{j+1} - i, N_{M_{j+1}}) \; c_i(k)}{\sum_{i=M_j}^{M_{j+1}-1} \alpha(M_{j+1} - i, N_{M_{j+1}})}$$

and

$$\varphi_j^+(k) = \frac{\sum_{i=M_j}^{M_{j+1}-1} \alpha(i - M_j + 1, N_{M_{j+1}}) \; c_i(k)}{\sum_{i=M_j}^{M_{j+1}-1} \alpha(i - M_j + 1, N_{M_{j+1}})},$$

with $N_{M_j} = M_j - M_{j-1} + 1$.

Euclidean distance between two segments weighted around to the marker is defined as:

$$\delta_j^E = \sum_{k=1}^{N_K} \left(\varphi_{j-1}^-(k) - \varphi_j^+(k) \right)^2$$

for $1 < j < N_\phi - 1$.

From this expression the fitness function is defined as the distance of the segmentation as the average:

$$\Gamma_\phi = \frac{1}{N_\phi - 2} \sum_{j=2}^{N_\phi - 1} \delta_j^E$$

It is easy to realize with some simple examples that this fitness function act as we expeceted.

E. Selection, Reproduction and Variation

There are several forms of performing the selection of the progenitors. As in Nature, it is not simply a matter of selecting the best. Individual selection is not related directly but probabilistically to its fitness. In terms of the search algorithm, the selection carries out the task of concentrating the computational effort in the regions of the space of solutions that are presented as more promising. In the evolutionary algorithm for speech segmentation (EASS) the *tournament* method was used. In this method ($v > 1$) individual are chosen thoroughly at random; they are made to compete by fitness and the winner is selected. This method is one of the most widely used due to its simple and efficient implementation.

The *reproduction* is the process through which the new population is obtained starting from selected individuals and variation operators. Crossovers and mutations were used in EASS. An additional variant in the reproduction that is not extracted directly from the biological evolution but that it is used with very good results is the *elitism.* This strategy keeps the best individual from previous population and copies it in the new population, independently of the selection and variation. In this way, the best solution through the generations is preserved. This strategy permits to increase the mutations probability and thus the solutions dispersion in the search space.

Mutation works altering gene values with a very low probability, for example $p_m = 0.001$. In conventional EC the Gaussian mutation is used, where the value of a gene is modified according to Gauss density probability function. A comparative review and combination of different mutation methods are dealt with in [13]. In EASS, for the selected individual a random chosen gene is mutated through the following equation:

$$\Delta M_{j*,G+1} = \Delta M_{j*,G} + R\, U(-1,1),$$

where j^* is the chosen gene for the mutation, G is the current generation number and R give the range in which the alteration is produced. The function $U(a,b)$ returns a real at random between a and b with an uniform distribution. The mutation is applied to each individual with probability p_m. If the probability were applied on each chromosome gene of all the individuals, it would be different. Strictly, this would be a probability of individual mutation and not a gene mutation probability.

The *crossover* is an operator that acts on two chromosomes to obtain other two. For EASS, two-point crossover is used. Crossover points are chosen at random but both chromosomes are cut in the same places. This ensures that chromosome length is kept after the crossover. However, as it was mentioned previously, it would be of interest for real time applications to have chromosomes with different number of segments and thus to choose a different point of crossover for each chromosomes.

F. Implementation details

All the programs were compiled with GNU C++, version egcs-2.90.29. The EO evolutionary computation toolkit[1] was used by subclassing. Speech signal processing routines were taken from ToFy function library[2], implemented for this purpose.

III. Results

Tests are split into two parts. In the first place are the tests that tend to show the most important characteristics of the algorithm. This is accomplished through a sequence of artificially created signals that contains information that results in an obvious segmentation. The second tests were performed on real speech files and the results are compared with a manual segmentation process and the segmentation accomplished by HMM in a speech recognition process. For all experiments a maximum number of generations of 500 was used.

A. Noise and harmonic

Noise and harmonic signal (N&H) of 1 second composed by mixed portions of silence, white noise and a 1000 Hz sine wave was generated. For this test file the EASS was applied with the parameters shown in first column of Table I.

TABLE I
Parameters used in experiments: where N is the number of individuals, GR is the gene range in ms, p_c is the crossover rate, p_m is the mutation rate.

Experiments →	N&H	Syllables	Words
N	10	200	200
GR [ms]	400	100	250
$N_\omega T_S$ [ms]	8	16	16
$N_D T_S$ [ms]	8	16	16
p_c	0.5	0.5	0.5
p_m	0.5	0.5	0.5
Elitism	yes	yes	yes

Figure 2 shows the resultant segmentation. These first results were attempted to emphasize some characteristic of the method and its elemental operation. The noise and harmonic signal are easily segmented, with a small computational load.

Moreover, as Table I shows, a minimal population is used. Although a toy example, this is even unusual in EC methods making the search nearest to the one accomplished by a gradient search method.

B. Speech Segmentation

In SS case, sentences from Albayzin [14] database are used. Several tests were run on speech utterances and the results always for the same file, with the objective of facilitating the analysis, are shown here.

In second column of Table I EASS parameters used in the first example of SS are shown.

[1] http://geneura.ugr.es/∼jmerelo/EO.html
[2] http://docentes.fi.uner.edu.ar/dhm/download/

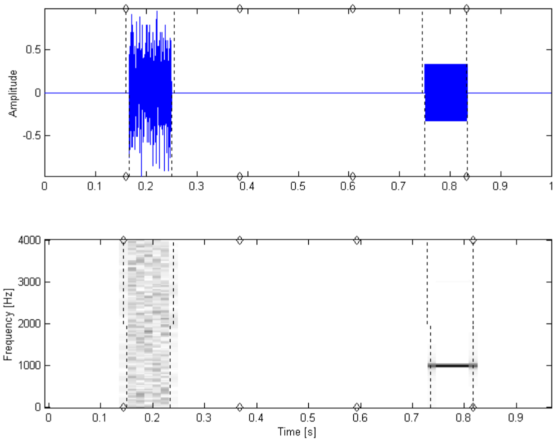

Fig. 2. Result of segmentation for "noise and harmonic" signal. The bottom half lines indicate the ideal segmentation while those of upper half indicate the segmentation accomplished by EASS. The rhombs ◊ indicate the linear segmentation.

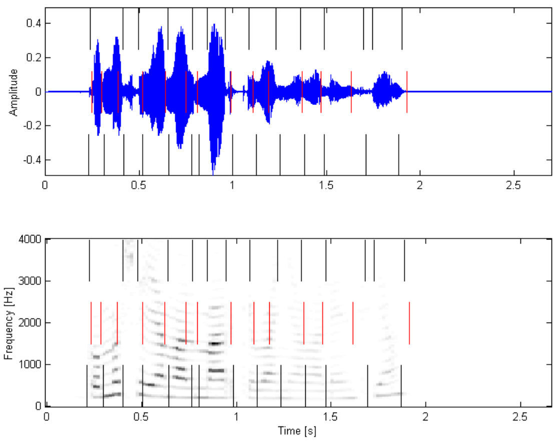

Fig. 3. Result of the syllable speech segmentation for the utterance /¿Cómo se llama el mar que baña Valencia?/ (What is the name of the sea that washes Valencia?). The bottom half lines indicate the ideal segmentation while top half lines indicate the segmentation accomplished by EASS. The short lines in the center indicate the syllable segmentation through HMM.

As many segments as syllables in the phrase were required. In Figure 3 the syllable segmentation result is observed. In several tests the convergence was obtained before 100 generations (∼17,2 seconds). However, a top number of 500 generations (∼86 seconds), are used in the following tests.

In the same graph (Figure 3) are observed the manual segmentation and the HMM segmentation as references. The manual segmentation was accomplished through energy and spectrogram analysis. The segmentation by HMM was obtained through continuous density models with 5 states for the phoneme and the silence (only 3 emitting) and 3 states for the short silence between words (1 emitting state). The parameterization used was mel cepstrum with delta, energy and elimination of average cepstral (a total of 26 parameters) [3]. The analysis window was 25 ms and the step of the analysis 10 ms, with Hamming window and pre-emphasis. The training is accomplished with 202 words in 600 extracted phrases from the Albaizin database. During the alignment of the models (to obtain the segmentation) is used a bigrammar with 206 nodes and 1137 arcs computed from all database.

The EASS parameters used in the second SS example are shown in third column of Table I. That is to say, only the range of the genes was modified, widening it to almost be able to include all the words. However, the exigency of quantity of segments according to syllable segmentation was kept. In Figure 4 it can be seen that the method tends to accomplish word segmentation.

In the syllables segmentation case (Figure 3) the markers found by the EASS segmentation coincide almost exactly (considering the analysis window size used) with the manual syllable segmentation markers and those of HMM. However, it can be seen that a mistake by omission is in the first syllable and one by insertion in the next-to-last.

In the segmentation of the Figure 4 it can be seen how the results are strongly modified by gene range choice. Guided by the typical lengths of the segments to find (syllables or words), the gene ranges can be selected. However, in speech, there are words that can be one syllable long (and to a phoneme, in some cases) and in the same way, many syllables can be a whole word long. This can be the weakest point of the method since other information relative to the valid words, the context and the grammar, as in the HMM case, is not used.

When tests were accomplished with the fast Fourier transform parameterization [3], the influence of energy conditioning the markers position, was observed. In this case, the markers were located in the maximums of signal energy variations, not segmenting syllables but rather delimiting vowels. This influence of signal energy for mel cepstrum parameterization, though in a minor degree, also was observed. However, energy is a good inter-words silence indicator. The tests done with linear prediction coefficients [3] do not differ much of the accomplished

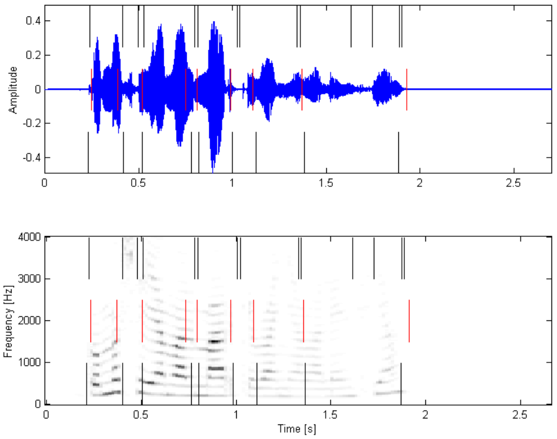

Fig. 4. Upon increasing the allele range (250 ms) the segmentation, that by the number of segments would have to be syllable, now tends be by words. The utterance is the same as in Figure 3.

with mel cepstrum but the computation of linear prediction coefficients is somewhat more slow.

IV. Discussion and Conclusions

A distinctive characteristic of the EASS method is that only one parameter determines its behavior as word or syllable SS: the gene range. Better results (and more easily interpretable) are obtained when a number of segments equal to the real speech segments are used. Besides, this dependency behavior with the range offers good perspectives for the implementation of the method with variable length chromosomes. This is not a complex technique but has the inconvenient of increasing the search space. The restriction of having to know the number of segments does not permit the application of the method to real-time SS. However, the segmentation of speech database, with known transcriptions, is an important application and this method is totally applicable to them. For real-time segmentation it would be necessary to work with variable length chromosomes (or other blind alternatives).

Other particularity of EASS method is that there is no training phase, neither parameters are stored for its subsequent utilization during the segmentation. Even though this causes the method to work with very little previous information on the task to accomplish, also endows it with robustness and flexibility, taking maximum advantage of the self-adaptive characteristic of EC algorithms.

In this first version of algorithm, several possibilities are left open to perform different improvements related to the evolution as well as in the signal processing, that is not necessarily restricted to speech. Some issues remains unexplored like the evaluation of better real-parameter recombination operators that should cause a better search. It is well known that performance of SS algorithms decay abruptly in the presence of noise, so another important issue to take into account is the evaluation of EASS robustness, and the neccesary modifications to improve its robustness. At this time we are performing a complete comparative study between EASS and more traditional approaches over an entire speech database.

References

[1] D. R. Reddy, “Segmentation of speech sounds,” *JASA*, pp. 307–312, 1966.

[2] J. Van Hemert, “Automatic segmentation of speech,” *IEEE Trans. on Signal Processing*, vol. 39, no. 4, pp. 1008–1012, April 1991.

[3] Deller. J. R., Proakis J. G., Hansen J. H., *Discrete-Time Processing of Speech Signals*, Prentice Hall, 1987.

[4] F. Brugnara, D. Falavigna, and M. Omologo, “Automatic segmentation and labeling of speech based on hidden markov models,” *Speech Communication*, vol. 12, no. 4, pp. 357–370, 1993.

[5] Vorstermans A., Martens J.-P. and Van Coile B., “Automatic Segmentation And Labelling Of Multi-Lingual Speech Data”, *Speech Communication*, Vol. 19, pp. 271-293, 1996.

[6] Jeong C. and Jeong H., “Automatic Phone Segmentation and Labelling of Continuous Speech”, *Speech Communication*, Vol. 20, pp. 291-311, 1996.

[7] Pauws S., Kamp Y. and Willens L., “A Hierarchical Method of Automatic Segmentation for Synthesis Applications,” *Speech Communications*, Vol. 19, pp. 207-220, 1996.

[8] Li T.-H. and Gibson J. D., “Speech Analysis and Segmentation by Parametric Filtering,” *IEEE Trans. On Speech and Audio Processing*, Vol. 4, No. 3, 1996.

[9] Bäck T., Hammel U. and Schewfel H-F., “Evolutionary Computation: Comments on History and Current State,” *IEEE Trans. on Evolutionary Computation*, Vol. 1, No. 1, Apr. 1997.

[10] Bhandarkar S. M. and Zhang H., “Image Segmentation using Evolutionary Computation”, *IEEE Trans. on Evolutionary Computation*, Vol. 3, No. 1, Apr. 1999.

[11] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Second Ed., Springer-Verlag, 1992.

[12] Merelo J. J., Carpio J., Castillo P., Rivas V. M., Romero G., Schoenauer M., “Evolving Objects”, *Third International Workshop on Frontiers in Evolutionary Algorithms*, Atlantic City, Feb. 2000.

[13] Chellapilla K., “Combining Mutation Operators in Evolutionary Programming,” *IEEE Trans. on Evolutionary Computation*, Vol. 2, No. 3, Sep. 1998.

[14] Casacuberta F., García R., Llisterri J. Nadeu C., Prado J. M. and Rubio A., “Development of a Spanish Corpora for the Speech Research”, *Workshop on International Co-operation and Standardisation of Speech Databases and Speech I/O Assessment Methods*, CEC DGXIII, ESCA and ESPRIT PROJECT 2589 “SAM”, Chiavari, pp. 26-28, Sep. 1991.