Abstract: - A new Neural Network architecture for real-time oriented speech denoising is proposed. It is based on Adaptive Spline neurons, whose peculiarity is the adaptive activation function. So, in the training phase, we can update both values of weights and activation function shape, obtaining networks with more flexibility and generalization capabilities. Net training is performed through the classical back-propagation rule. We focused our attention to continuous uncorrelated disturbs and we tried two kinds of approach: in the first one we processed the whole signal by a single network, while in the second one we operated a frequency sub-bands decomposition and we processed every sub-channel separately in a parallel way. The first approach is less heavy but the second one gives better results, due to the fact that, in practical application, background noise is frequency dependent. Results show improvements of Signal to Noise Ratio (SNR) and better performances in comparison with classical denoising neural networks.

Key-Words: - Speech Enhancement, Noise Reduction, Adaptive Filters

1 Introduction
The increasing development of communication and recording systems raises the need of improving performances in speech denoising. One of the most important goal is to improve noise-corrupted speech intelligibility in man to man or man-machine communication. Performances of modern automatic speech recognition systems, for example, are greatly affected by the presence of noise, so that correct speech recognition in noisy environments is often difficult or quite impossible.

There are several kinds of speech noises, such as impulsive, continuous, correlated or uncorrelated ones. It is obvious that a general solution to the problem of speech denoising is not achievable; a specific noise-oriented solution is needed. In this research we focused our attention to continuous uncorrelated disturbs (background noise).

The classical approach to the problem, based on Spectral Subtraction, is still one of the most employed one; its main fault is due to residual "musical noise" which causes metallic voice distortion [1,2]. In recent years, neural networks were successfully applied in audio signal processing. The principal disadvantage of this approach is due to the great size of implemented structures and to the low generalization capabilities.

The aim of our research is to obtain low size real-time oriented neural architectures using Adaptive Spline Neural Network (ASNN) for improvement of SNR in background-noise corrupted speech signals. Comparison is made with classical Multi Layer Perceptron (MLP) neural networks used in denoising applications [3,4].

2 Spline Networks
Traditional multilayer feed-forward neural networks are based on a model of biological neurons. Each neuron computes the weighted sum of its inputs and applies to this sum a non-linear function called activation function; generally, an activation function such as $a(1-e^{-bx})/(1+e^{-bx})$ is used. Structures with multiple layers guarantee approximation capability of complex functions of several inputs.

Recently, a new interest in adaptive activation functions has arisen; the simplest of them consists in involving polynomial functions, which allow to increase the neuron complexity and to reduce the size of the network. This solution implies some drawbacks, principally with the adaptation of the coefficients in the learning phase, due to spurious minima and maxima. In addition, a polynomial function is a non-bounded function and the resulting approximation is generally poorly smooth. Later, adaptive spline activation function was introduced [5,6]. In this approach, each neuron is characterized by a different activation function whose shape can be modified through some control points. Further works demonstrated that such neuron architecture could improve approximation and generalization abilities of the entire network. The major advantage in this approach in given by the possibility to implement activation functions using cubic spline interpolation of control points. So in every training step we can update both activation function shape and input weights values, obtaining networks with more flexibility and generalization capabilities.

In the following, we report a review of spline network theory.
2.1 Spline Curve

A planar spline curve is a two-dimensional vector, whose components are piecewise polynomial, univariate function of the same degree: its mathematical formulation ensures both its continuity and the existence of its derivatives, along the curve and in correspondence to the joining points between the various curve spans. A general expression for that curve would be:

\[ F(u) = \left[ F_x(u) \quad F_y(u) \right]^T = \sum_{i=0}^{N-1} C_i F(u) \]

(1)

where \( C \) is the concatenation operator and \( F_i(u) \) the \( i \)-th curve span (or patch). The indices of the \( C \) operator in (1) are valid only for cubic polynomials. The choice of using cubic polynomials was made because of the tradeoffs between the requested properties and computational complexity.

The parameter \( u \) has the property of being local and its domain is \( 0 \leq u \leq 1 \) for every curve span. Hence, there must be a unique mapping that allows us to calculate the local parameter \( u \), as well as the proper curve span \( i \), from the abscissa global parameter. In this way, we can represent any point lying on the spline curve \( F(u) \) as a point belongs to the single \( F_i(u) \) curve span. It follows (see [5]) that the \( i \)-th curve span can be described as follows:

\[ F_i(u) = \left[ F_{x,i}(u) \quad F_{y,i}(u) \right]^T = \sum_{j=0}^{3} Q_{i,j} C_j(u) \]

(2)

were:

a) \( Q \) are called control points and are represented by the \( x \) and \( y \) coordinates \( Q = [x_{q,i}, y_{q,i}] \) with the constriction that:

\[ q_{i,0} < q_{i,1} < \cdots < q_{i,N} \]

b) to avoid loops (reverse ordering of abscissa) or multiple output values for a single abscissa (overlapping abscissa).

c) \( C_j(u) \) are the spline polynomials:

\[
\begin{align*}
C_0(u) &= \frac{1}{2}(-u^3 + 2u^2 - u) \\
C_1(u) &= \frac{1}{2}(3u^3 - 5u^2 + 2) \\
C_2(u) &= \frac{1}{2}(-3u^3 + 4u^2 + u) \\
C_3(u) &= \frac{1}{2}(u^3 - u^2)
\end{align*}
\]

2.2 The SG (Sigmoid Generalized) Neuron

The choice of a parametric curve to implement activation function implies that, once computed the output of weighted sum \( x'^L \), it needs to find the correspondence in the parametric curve (that is to determine the span of the curve) and then to map the value through the curve.

In Figure 1 a SG neuron is shown; in the same figure we describe also how the output of \( k \)-th neuron in the \( l \)-th layer of a net is obtained. The block SG1 represents the inversion of the \( x \)-axis component of the parametric spline function. By choosing uniform distribution of samples along \( x \)-axis:

\[ F_u(u) = u \Delta x + d_{y,i+1} \]

we can easily make the inversion of axis without calculating the root of third order polynomial function, with the simple assumption:

\[
\begin{align*}
z_i^{(l)} &= \frac{s_i^{(l)}}{\Delta x} + \frac{n - 2}{2} \\
d_i^{(l)} &= \left\lfloor z_i^{(l)} \right\rfloor \\
u_i^{(l)} &= z_i^{(l)} - \left\lfloor z_i^{(l)} \right\rfloor
\end{align*}
\]

(3)

were \( \lfloor \cdot \rfloor \) is the floor operator, \( n \) is the number of control points, \( z_i^{(l)} \) is an internal dummy variable, \( a_i^{(l)} \) and \( u_i^{(l)} \) are respectively the index of considered span and the respective internal parameter \( u \). For example, \( a=0 \) corresponds to the first span of the curve, so the first four control point \( \{q_{i,0}, q_{i,1}, q_{i,2}, q_{i,3}\} \) are considered from a Look Up Table (LUT) were are saved all the control point \( \{q_{i,0}, \ldots, q_{i,N-1}\} \), as shown in Figure 2.
The block SG2 represents the (2) that can be easily rewritten as:

\[
F(u) = \begin{bmatrix} u^3 & u^2 & u \end{bmatrix}^T \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_{0,0} \\ Q_{0,1} \\ Q_{0,2} \\ Q_{0,3} \end{bmatrix}
\]

for Catmull-Rom spline, where \(F(u)\) corresponds to the span \(a\) calculated in block SG1 with the (3).

Back-propagation learning algorithms are used to train both weights and control point of network.

### 3 Employed Structure

The developed network is a three layer network with a single neuron output layer. Network inputs are the noisy signal samples. There is an upper limit to the input size due to the fact that speech signal can be considered stationary for time intervals shorter than 15 ms, corresponding to 120 samples for sample frequency \(F_s=8\) KHz. However, in practical application a shorter pattern of inputs is enough, so we considered an input size of 40 samples. Several configurations were studied, varying the number of neurons employed in input and hidden layers and also trying multiple out structures with more than one neuron in output layer. Good results were achieved with low size structures employing only three neurons in input and hidden layers and with one single output neuron (3:3:1 structure), as shown in Figure 3.

This structure gets 40 samples of noisy speech signal and gives a single reconstructed output sample of clean estimated signal, as shown in Figure 4. The net then shifts one single position and reconstructs the next clean sample from the next set of 40 inputs. The process continues until signal end is reached.

Beside this structure, which operates directly on noisy signal samples, an alternative one based on sub-bands approach is proposed, as described in the following section.

### 4 Sub-Bands Approach

In practical applications, background noise couldn't be considered a white noise because its energy spectral distribution is not constant. In this situation, signal sub-band decomposition could improve denoising performances because it makes possible to operate strongly on noisiest sub-bands and weakly on the other ones.

![Figure 5: The employed sub-band structure.](image)

Two kinds of different filter banks were implemented [7,8], in way of getting linear and logarithmic sub-bands decomposition. Uniform sub-bands decomposition is achieved applying the known Pseudo-Quadrature Mirror Filter Banks Near-Perfect Reconstruction. The signal passes through an analysis filter bank that splits it in \(M\) sub-channels. Each channel is decimated by a factor \(M\) and processed by a neural structure. Filter bank operates directly on time domain so that sub-band output signals are time-domain samples. The whole output signal is then obtained interpolating the \(M\) sub-signals and passing them through a synthesis filter bank. Logarithmic sub-bands decomposition is achieved with a Quadrature Mirror Filter that splits the input signal into both its high and low frequency sub-bands. Lower frequency sub-band is recursively passed through the filter until the desired logarithmic decomposition is achieved. Signal reconstruction is then obtained with a dual scheme. This decomposition is very useful in many practical applications because real environment noise usually has higher energy at lower frequencies.

The block scheme of the structure used for implementing the sub-bands method is reported in Figure 5. As this Figure shows, in the sub-bands approach every band of the original sound could be processed by a single network in a parallel way. Moreover, also the training phase is made in parallel in all channel through a noise that is peculiar of the channel frequency band.
5 Net Training
The training phase needs the availability of original clean waveforms together with their noisy versions. The used learning algorithm is based on the classical back-propagation rule: at every processing step, clean and noisy signals are compared and the error between the clean and estimated samples is calculated. Network weights and control points are then rearranged by minimizing instantaneous Mean Square Error. Network parameters are re-arranged at every single computation step, so we have 8000 parameter updates per second at \( F_s = 8 \text{ KHz} \). The first training cycle ends when input signal end is reached. Decreasing exponential learning rate was employed as to get stability and fast convergence capabilities. The training ends when not relevant parameter changes are achieved.

In sub-band approach, all channels are processed simultaneously and the comparison is executed separately for each channel.

6 Simulation Results
Different speech samples with different speaker and different utterance were considered in training computation. Real environment sampled noise was employed and noisy signals at different SNR were processed. Used sampling frequency was \( F_s = 8 \text{ KHz} \). Both single-band and sub-band architectures were tested on training signals (Training Set) and on other signals not employed in training process (Test Set). Sub-bands number was 8 in either linear and logarithmic approach. The developed architectures show better SNR improvements in comparison with classical perceptron neural approach [3,4]. SNR improvement of about 10 dB is achieved for logarithmic sub-bands decomposition in training test signal (starting SNR=7dB), as shown in Figure 6, and of about 7 dB in Test Set signals (starting SNR of 5,7,10,15). A presence of residual distortion was however found in test signals. Proposed architectures have lower complexity as to what concerns net dimensions and topology.

![Figure 6: SNR improvement: comparison with classical neural networks (MLP).](image)

Classical Perceptron Neural Networks employ a higher number of neurons (up to 180 in 60:60:60 structure), moreover providing lower performances, as shown in Figure 6. In proposed single band spline architecture, the number of neurons is only 7. In sub-band approach, this number should be multiplied by sub-bands number, but even in this case overall dimensions are still lower than classical approach ones.

7 Conclusion
A new neural architecture for speech signal denoising was developed. An Adaptive Spline neuron model was employed in order to improve network capabilities, and a real additive environment noise was considered. Because the real environment noise is, in general, frequency dependent, we propose also a sub-band decomposition method, in which both learning and operative phases can be implemented by parallel computation, with time saving. Results show better performances compared to classical MLP neural networks, with SNR improvements up to 10 dB. The developed architecture is real-time application oriented because of its very low complexity.

References: