

Generalized Splitting 2D Flexible Activation Function

Francesca Vitagliano, Raffaele Parisi, and
Aurelio Uncini

INFOCOM Dept. - University of Rome “La Sapienza”,
Via Eudossiana 18, I-00184 Rome, Italy,

aurel@ieee.org,

WWW home page: <http://infocom.uniroma1.it/aurel>

Abstract. It is well known that in problems where both amplitude and phase recovery is essential - like in signal processing for communications, or in problems of nonlinear signal distortions, like control, signal processing and imaging applications - it is important to consider the complex nature (and thus the intimate relation between real and imaginary part) of the data.

One of the main problem to design complex neural networks (CpxNN) consists in the definition of the complex Activation Functions (AF): to ensure the universal approximation network capabilities, the AFs should be bounded and differentiable. In the complex domain these characteristics are in contrast with Liouville’s theorem, which asserts that the only bounded and differentiable (analytic) function is the constant function. In this paper we investigate the use of 2D spline to define a new class of flexible activation functions, which are bounded and (locally) analytic suitable to define a new class of complex domain neural networks (CpxNN).

1 Introduction

The classical neuron computes the weighted sum of its inputs and feeds it into a nonlinear function called activation function (AF) [1]. The behavior of a neural network (NN) built with such neurons, as in the multi-layer perceptron (MLP), thus depends on the chosen AFs. Sigmoids are commonly used for this purpose. Different classes of nonlinear AFs, depending on some free parameters, have also been widely studied and applied (see for example [2]).

Although the main theoretical developments are defined for real valued NNs, it is well known that complex domain NNs (CpxNN) are suitable for many signal processing applications.

It is well known that using CpxNN is more advantageous than using a real-valued NN fed with a pair of real numbers [3] - [5].

In CpxNNs, one of the main problem is related to the complex domain AF. Let $f(z)$ be the complex AF with $z \in \mathbb{C}$ (z is the complex linear combiner output); the main properties that $f(z)$ should satisfy are:

- 1) $f(z) = u(x, y) + jv(x, y)$;
- 2) $f(z)$ should be non linear and bounded;
- 3) In order to derive the backpropagation (BP) algorithm the partial derivatives of $f(z)$ should exist and be bounded.

Unfortunately the main difficulty is the lack of bounded and at the same time analytic complex nonlinear AFs in the complex plane \mathbb{C} . In fact Liouville's theorem (see [10] for more details) states that: *'The only bounded differentiable (analytic) functions defined for the entire complex domain are constant functions'*.

Clearly the properties of boundedness and differentiability in all the complex domain are contrasting requirements, if we want to use a complex AF defined in \mathbb{C} . In other words $f(z)$ should be defined as a nonlinear complex function that is bounded almost everywhere in the complex domain \mathbb{C} .

Recently, a complex-valued adaptive spline neural network has been presented [12]. The author used the splitting method, where the two real functions are substituted by two flexible spline curves controlled by a small number of parameters. It is shown that this architecture is well suited for supervised signal processing applications, because it is based on an efficient Catmull-Rom spline AF whose regularization properties are described in [13].

In this paper we introduce the use of bi-dimensional (2D) spline in order to define a new class of flexible AF, called generalized split AF (GSAF), which are bounded and (locally) analytic functions, suitable to define a new class of complex domain neural networks. In order to demonstrate the effectiveness of the proposed model, experiments on channel equalizations are described in the last Section.

2 On the Complex Activation Functions

In the last years several studies have been carried out in order to develop a complex domain learning algorithm using suitable AFs. Kim and Guest [3], proposed a complex domain modification of the BP algorithm by an extension of the real valued sigmoidal activation function to the complex plane. However, this function has a periodic singularity:

$$\frac{1}{1 + e^{-z}} \rightarrow \infty \quad \text{for } z = \pm j(2k + 1)\pi \quad \forall k \in \mathbb{N} \quad (1)$$

In [4] Clarke proposed to use a hyperbolic function ($\tanh z$) in the complex domain as a generalization of real valued activation functions.

In order to have a bounded and differentiable AF, Benvenuto *et al.* and Leung and Haykin [5]-[7], proposed an *ad-hoc* extension of the real valued backpropagation. According to properties 2) and 3) previously stated, they proposed a *split* activation function consisting of the superposition of a real and an imaginary (I and Q signal components) part AF

$$f(z) = f_I(\Re(z)) + jf_Q(\Im(z)) ; \quad (2)$$

where functions $f_I(\cdot)$ and $f_Q(\cdot)$, can be simple real-valued sigmoids or more sophisticated adaptive functions. However such split activation function is not analytic and the BP from the output layer takes split paths through disjoint real-valued gradients.

In order to develop a fully complex domain BP Georgiou and Koutsougeras in [8] proposed the AF defined as

$$f(z) = \frac{z}{c + \frac{1}{r}|z|} ; \quad (3)$$

where c and r are suitable constants. This function maps the complex signals into an open circle of radius r . By this way the activation function takes only values belonging to the interval $(0, r]$.

Hirose in [9] proposed the use of a fully complex AF defined as

$$f(s_k e^{j\beta_k}) = \tanh(s_k/m) e^{j\beta_k} ; \quad (4)$$

where s_k and β_k are the norm and the argument of the summation of the input vector fed from the previous hidden layer ($j\beta_k = \sum_j w_{kj} y_j$), and m is a constant which is inversely related to the gradient of the absolute function value $|f|$ along the radius direction around the origin of the complex coordinate.

The *suitable constants* of the AFs (3) and (4) are chosen in order to normalize or to scale the amplitude of the complex input signal. However, these functions preserve the phase, thus being incapable of learning the phase variation between the input and the target in NN without delay lines at the input layer. Moreover due to their radial mapping, as stated in [15], in the case of time-delayed NN they perform poorly in restoring nonlinear amplitude and phase distortion of non constant modulus signals.

A. Elementary Transcendental Functions

More recently Kim and Adali, in order to define a family of useful fully complex AFs, proposed the use of the so called elementary transcendental functions (ETF) [15].

They reduced the 'desirable properties' of a complex AF to the unique condition:

$$f'(z) = f_x = -i f_y \quad (5)$$

that is the Cauchy-Riemann equation.

They classified the ETFs in two categories of unbounded functions, depending on which kind of singularities they possessed. A singularity is a point in which a function is not analytic (and thus not differentiable): if $\lim_{z \rightarrow z_0} f(z) \rightarrow \infty$ but the function is analytic in a deleted neighborhood of z_0 (that is a pole), the singularity is said to be *isolated*; if $\lim_{z \rightarrow z_0} f(z)$ exists it is isolated but *removable*; if none of these cases are met, the function has an *isolated essential singularity*.

The proposed elementary functions including $\tan(z)$, $\operatorname{atan}(z)$, $\sin(z)$, $\operatorname{asin}(z)$, $\operatorname{acos}(z)$, $\tanh(z)$, $\operatorname{atanh}(z)$, etc. are identified to provide adequate nonlinear discriminant capabilities as required for an AF.

These transcendental functions are entire (analytic) and bounded *almost everywhere*, i.e. they are unbounded only on a set of points having zero measure. If used as AFs in neural networks they assure convergence almost everywhere (with probability 1).

Moreover, in signal processing applications where the domain of interest is a bounded neighborhood of the unit circle these singular points scarcely pose a problem.

B. Universal approximation property

Recently in [16] the universal approximation theorem for fully CpxNNs has been demonstrated.

Let I_n denote the n -dimensional unit cube $[0, 1]^n$ and $C(I_n) \subset M(I_n)$ the space of all continuous complex functions on I_n . The finite sums of the form

$$G(z) = \sum_{k=1}^N \beta_k f(w_k^T z + \theta_k) ; \quad (6)$$

where $\beta_k, \theta_k \in \mathbb{C}$, $w_k, z \in \mathbb{C}^n$, N is the number of hidden units and $f(z) : \mathbb{C}^n \rightarrow \mathbb{C}$ are dense in $C_1(I_n)$ if $f(z)$ is a complex bounded measurable discriminatory function i.e.

$$\forall g \in C(I_n), \forall \varepsilon > 0, \forall z \in C(I_n) \Rightarrow \exists G(z) \mid G(z) - g(z) \mid < \varepsilon \quad (7)$$

where $f(z)$ is an activation function of the hidden layer and $G(z)$ is an output of the net.

In [16] they also demonstrated the theorem for any complex bounded measurable discriminatory function and further for functions having isolated or essential singularities, but in compact subsets of the deleted neighborhood of the singularity.

All these functions are bounded almost everywhere and have countable singular points; some of them can be handled separately during the learning process (removable singularities) or can be placed outside of the working domain. In this way the conflicting requirements between boundedness and differentiability can be relaxed by paying attention to the domain of operation, that can be easily identified for almost all applications.

3 Generalized Split AF by 2D Spline

This section is dedicated to the description of the new AF starting from the theories on fully CpxNNs: starting from previously reported considerations we tried to employ a generalization of the split method proposed in [12].

If we consider the expression of a fully complex function in relation to the real and imaginary part we may write:

$$f(z) = f(x, y) = u(x, y) + jv(x, y) \quad (8)$$

in which, as we have shown in the introduction, the two functions must be bounded and differentiable at least in the whole domain of the problem at hand.

If we consider each part as a function of two variables, we can perform each of them with a bi-dimensional spline; one plays the role of the real part and one of the imaginary part of the complex activation function. With regard to the 'desired properties' stated for the fully complex AFs we can note that 2D splines:

- They are non linear functions with respect to the coordinates; thus $f(z)$ is a nonlinear function with respect to x and y ;
- They haven't singularities and they are bounded for each $z = x + jy$;
- The partial derivatives u_x, u_y, v_x, v_y are continuous and bounded;
- The condition $u_x v_y \neq v_x u_y$ is verified;

If we consider Adali's conditions, the Cauchy-Riemann equations are not satisfied by the complex 2D spline AF itself, but we tried to impose them by an algorithm constraint during the learning process:

$$u_x = v_y = \frac{(u_x + v_y)}{2}, \quad u_y = -v_x = \frac{(v_x + u_y)}{2}; \quad (9)$$

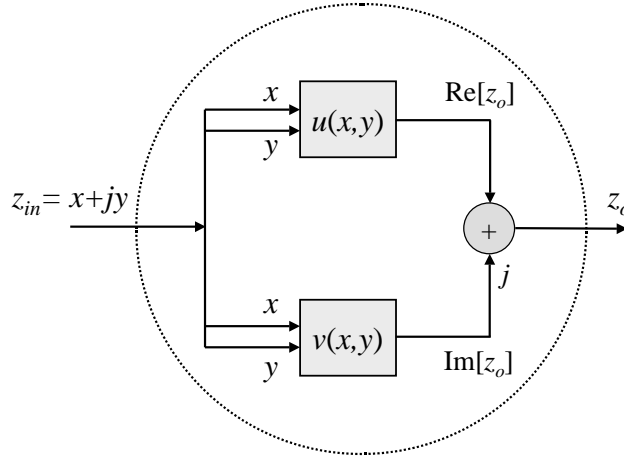


Fig. 1. Generalized splitting activation function. The functions $u(x, y)$ and $v(x, y)$ are built using flexible bi-dimensional splines.

The neural architecture apparently is the same as a traditional MLP, but the difference lays inside each neuron.

◇ *Backpropagation for the generalized splitting AF*

Clearly the input to the neuron is the complex weighted sum of the outputs of the previous layer ($l - 1$) of the net:

$$z_{in} = x + jy = \sum_{k=0}^{N_{l-1}} w_k x_k = \sum_{k=0}^{N_{l-1}} (w_{I,k} + jw_{Q,k}) (x_{I,k} + jx_{Q,k}) \quad (10)$$

When a signal z_{in} comes to the input of the neuron, both real and imaginary part are sent to the two functions, which are each a bi-dimensional spline. The output of each function is a real value; we impose these two outputs to be respectively the real and imaginary part of the output of the whole activation function. This pair is met at the output of a neuron, is weighted with complex weights and summed with all the other weighted outputs, ready to be sent to the next layer. Consequently the BP algorithm is fully complex.

Let $e(n) = d(n) - y(n)$, during the backward pass the δ 's are calculated as follows:

$$J = \frac{1}{2} \sum_{j=0}^{N_M} |e_j|^2; \quad (11)$$

$$\delta = -\frac{\partial J}{\partial u} - j \frac{\partial J}{\partial v}; \quad (12)$$

where for a neuron of the output layer ($M - 1$) it follows that

$$\begin{aligned} \frac{\partial J}{\partial u_k} &= \frac{1}{2} \frac{\partial}{\partial u_k} \sum_{j=0}^{N_M} |e_j|^2 = e_{I,k} \frac{\partial}{\partial u_k} \sum_{j=0}^{N_M} (d_j - f_j(z)) = -e_{I,k} \\ \frac{\partial J}{\partial v_k} &= \frac{1}{2} \frac{\partial}{\partial v_k} \sum_{j=0}^{N_M} |e_j|^2 = e_{Q,k} \frac{\partial}{\partial v_k} \sum_{j=0}^{N_M} (d_j - f_j(z)) = -e_{Q,k} \end{aligned} \quad (13)$$

and for a hidden layer (l):

$$\begin{aligned} \frac{\partial J}{\partial u_m} &= \sum_{i=0}^{l+1} \frac{\partial J}{\partial u_i} \left(\frac{\partial u_i}{\partial x_i} \frac{\partial x_i}{\partial u_m} + \frac{\partial u_i}{\partial y_i} \frac{\partial y_i}{\partial u_m} \right) \\ \frac{\partial J}{\partial v_m} &= \sum_{i=0}^{l+1} \frac{\partial J}{\partial v_i} \left(\frac{\partial v_i}{\partial x_i} \frac{\partial x_i}{\partial v_m} + \frac{\partial v_i}{\partial y_i} \frac{\partial y_i}{\partial v_m} \right) \end{aligned} \quad (14)$$

We called $x_i + jy_i$ the generic input to layer ($l + 1$), meaning by that, as before, the weighted sum of the outputs of the previous layer.

This new architecture differs from the network with real 2D splines in two main aspects:

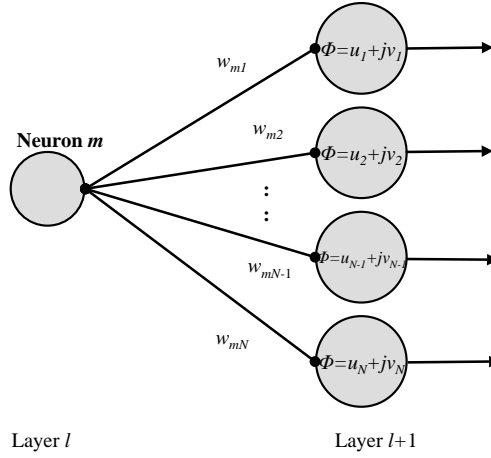


Fig. 2. Connection between a neuron of a layer and all the neurons in the following layer.

- When used to process complex signals, the real 2D spline considers the real and imaginary part of an input as two distinct signals; they are fed two times to each input neuron (a neuron j in the input layer has two inputs $s_{1,j}$ and $s_{2,j}$) but with different weights; as a consequence the number of free parameters is higher than in a complex S.N.
- The output of an activation function of the real 2D spline is single and real; no more trace of the complex nature of the data is held, thus forcing the use of a pseudo-complex backpropagation.

4 Experimental Results

In order to proof the effectiveness of the generalized splitting AF CpxNN and of its adaptation scheme, several experiments, consisting in communication channel equalization using different channels model have been carried out.

The first experiment consists in the equalization of a non-minimum-phase linear channel modelled by an impulse response $h=(0.3482, 0.8704, 0.3482)$ [17]. At the filter's output is applied a polynomial memoryless nonlinear function f , as shown in Figure 3.

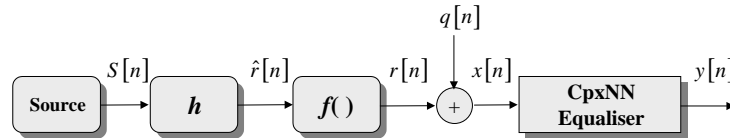


Fig. 3. Discrete time model of a digital communication system.

The output of the channel can be simply computed as

$$\hat{r}[n] = \sum_{k=0}^{L-1} h[k]s[n-k] \quad (15)$$

where the memoryless nonlinear function is quadratic polynomial

$$r[n] = \hat{r}[n] + 0.2\hat{r}^2[n] \quad (16)$$

and finally, by adding Gaussian white noise $q[n]$ with variance σ_N^2

$$x[n] = r[n] + q[n] \quad (17)$$

such that the signal to noise ratio (SNR) is defined as

$$SNR = \frac{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N E[|r[n]|^2]}{\sigma_N^2} \quad (18)$$

where $E[\cdot]$ represents the expectation operator.

The waveform is a 4-QAM (quadrature amplitude modulation) whose alphabet is the set: $\frac{1}{\sqrt{2}}\{(1, j); (1, -j); (-1, -j); (-1, j)\}$.

The network is composed of two layers: the hidden has two neurons with complex multidimensional spline activation functions, while the output is linear.

In figure 4 is shown the symbol error rate (SER), in relation to the SNR; both are expressed in dB.

The network was compared with a real 2D spline with two neurons in the hidden layer and linear output, and with a MLP network with 20 hidden neurons and linear output.

The number of free parameters for NN_20 is 302, for Sp_2D_2 is 88 and for CpxSp2D_2 is 98.

As we can see from the table, the performances of the complex network are better than that of the other two networks.

In the second experiment the complex spline network is compared with a fully complex network using $\tanh(z)$ as AF and with a split (real) network using $\tanh(x)$. The problem is still channel equalization, but with a different channel model and a QPSK signal of alphabet $\{(1, 0); (0, j); (-1, 0); (0, -j)\}$.

The channel is modelled by a complex FIR filter defined by the following transfer function

$$H(z) = \frac{\sqrt{2}}{2}(1 + j) + 0.1z^{-1} \quad (19)$$

while the nonlinear part is a third order component providing a nonlinear rotation of the constellation points

$$r[n] = \hat{r}[n] + 0.2\hat{r}^3[n] \quad (20)$$

White Gaussian noise, with independent real and imaginary parts, with zero mean and variance σ_N^2 is added at the channel output.

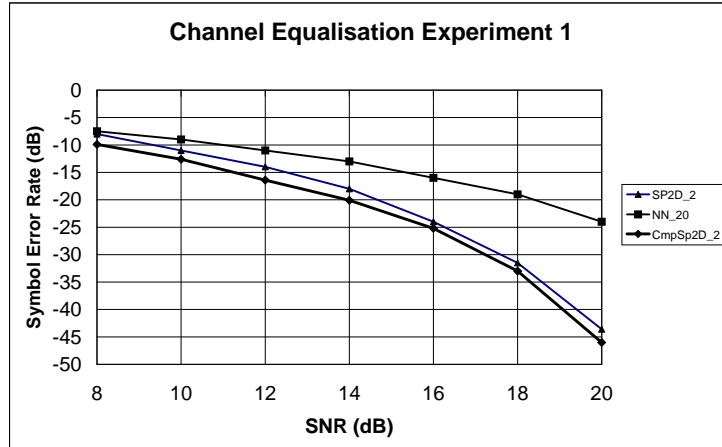


Fig. 4. Symbol error rate vs signal to noise ratio for the nonlinear channel $H(z)$ with a 4-QAM signal.

The complex spline network is composed of two layers: the hidden has only one spline neuron and the output is linear, for a total of 50 free parameters. The other two networks have both 5 neurons in the hidden layer and linear output, for a total of 50 free parameters. The networks are trained with 5×10^3 symbols (independent and uniformly distributed), while the test set is composed of 3×10^6 symbols. For the spline network the learning rate was fixed to $\mu = 0.001$ and the spline rate was $\eta_{sp} = 0.01$ in all the trials. The tests were repeated for ten distinct realizations, each with different initialization (the free parameters were initialized randomly as in the previous experiments).

Figure 5 reports the results of the performed experiment. As we can note the spline complex network reaches better performances with respect to tanh networks.

It must be pointed out that in simulations with complex tanh as AF, if no limitation is imposed to the range swept away by the function, the learning is discontinuous and for low levels of SNR the algorithm may not converge at all. Moreover, the weights are constrained to be initialized to small values, otherwise the updating is oscillating. The learning process is very sensitive to the size of the learning rate, constraining it to very small values and it must be changed during the training process (by trials). As a consequence, for low levels of SNR, the convergence is very slow and very often the target MSE (prefixed for each SNR test and reached by the spline network) is not reached (3 or 4 dB less).

5 Conclusions

In this work we introduce a new complex activation function for neural networks, based on spline interpolation and in order to process complex data. Only in

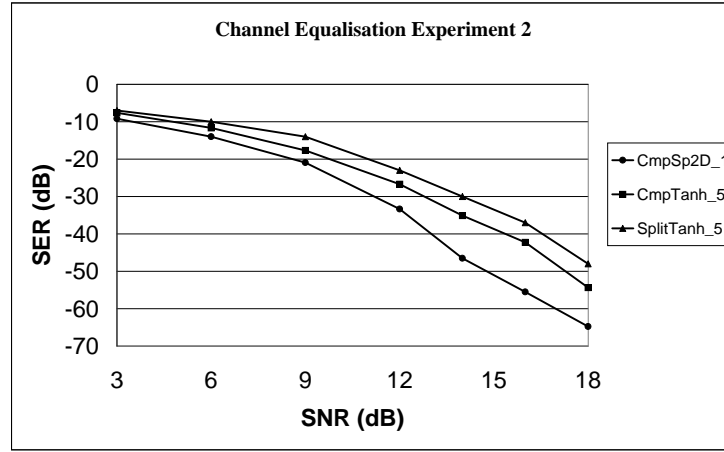


Fig. 5. Symbol error rate vs signal to noise ratio for the nonlinear channel $H(z)$ with a QPSK signal.

recent time efforts were made to find a way of handle these kind of signals: in fact previously the traditional approach considered complex signals just like a pair of real numbers and real and imaginary parts were elaborated separately as independent from each other. This approach doesn't take advantage of the deep correlation between real and imaginary part of a complex signal and represents a compromise to avoid the difficulty to find nonlinear complex activation functions. In fact the main obstacle to their use is the conflict between the boundedness and the differentiability of complex functions in the whole complex plane.

Another approach is that of using fully complex neural networks with activation function that are differentiable and bounded almost everywhere. We think that fully complex neural networks are best suited to deal with complex data because they are defined in the same domain (the complex plane) and provide substantial advantages in learning complex nonlinear mappings because of their efficiency. The experimental results confirm the validity of this approach. In particular complex bidimensional spline networks outrun the performances of other complex networks in both generalization capability and speed of convergence. This is due to the desirable properties owned by spline functions, that demonstrated to be a very powerful tool.

However this work is only a preliminary step in the field of complex networks and further investigation may lead to new discoveries and better improvements.

References

1. Haykin S. "Neural Networks, a comprehensive foundation" second edition, Prentice Hall International Inc., New Jersey, 1999.

2. K. Hornik, M. Stinchcombe, H. White, "Universal Approximation of an Unknown Mapping and Its Derivatives Using Multilayer Feedforward Networks", Neural Networks, Vol. 3, pp. 551-560, 1990.
3. M.S. Kim and C.C. Guest, "Modification of Backpropagation Networks for Complex-Valued Signal Processing in Frequency Domain", Proc. of Int. Joint Conference Neural Networks, San Diego (CA), pp. II 27-31, June 1990.
4. T.L. Clarke, "Generalization of Neural Networks in the Complex Plane", ", Proc. of Int. Joint Conference Neural Networks, San Diego (CA), pp. II 435-440, June 1990.
5. N. Benvenuto, M. Marchesi, F. Piazza, A. Uncini, "A Comparison Between Real and Complex Valued Neural Networks in Communication Application" Proc. of INNOC 91, European Int. Neural Network Conference, Helsinki (Finland), June 1991.
6. N. Benvenuto, F. Piazza, "On the Complex Backpropagation Algorithm", IEEE Trans Signal Processing, Vol.40, pp.967-969, Apr. 1992.
7. H. Leung, S. Haykin, "The Complex Backpropagation Algorithm", IEEE Trans Acoust. Speech and Signal Process., Vol. ASSP-39, pp.2101-2104, Sept. 1991.
8. G. Georgiou and C. Koutsougeras, "Complex Backpropagation", IEEE Trans. On Circuits and Systems II, Vol. 39, No. 5, pp. 330-334, May 1992.
9. A. Hirose, "Continuous Complex-Valued Back-propagation Learning", Electronics Letter, Vol. 28, No. 20, pp. 1854-1855, September 1992.
10. W. Rudin, "Real and Complex Analysis", McGraw Hill, 1974.
11. N. Benvenuto, M. Marchesi, F. Piazza, A. Uncini, "Non Linear Satellite Radio Links Equalized Using Blind Neural Networks", IEEE Int. Conference on Acoustic Speech and Signal Processing, Toronto, Canada, pp. 1521-1524, May 1991.
12. A. Uncini, L. Vecchi, P. Campolucci, F. Piazza, "Complex-Valued Neural Networks with Adaptive Spline Activation Function for Digital Radio Links Nonlinear Equalisation", IEEE Transaction On Signal Processing, Vol.47, No 2., 1999.
13. L. Vecchi, F. Piazza, A. Uncini, "Learning and Approximation Capabilities of Adaptive Spline Activation Function Neural Networks", Neural Networks, Vol. XI, No.2, pp. 259-279, 1998.
14. Solazzi M. Uncini A. "Regularising Neural Networks Using Flexible Multivariate Activation Function", submitted to Neural Networks, 2003.
15. T. Kim, T. Adali, "Complex Backpropagation Neural Networks Using Elementary Transcendental Activation Functions", Proc. of IEEE ICASSP, Vol. II, pp.1281-1284, 2001.
16. T. Kim, T. Adali, "Universal Approximation of Fully Complex Feed-Forward Neural Networks", Proc. of the IEEE ICASSP, Vol. I, pp.973-976, 2002.
17. [17] Proakis J.G. "Digital communications" second edition, McGraw-Hill International, New York. [1989]