Applied Computational Electromagnetics Society Journal

Special Issue on
Neural Network Applications in Electromagnetics

Guest Editor
Christos Christodoulou
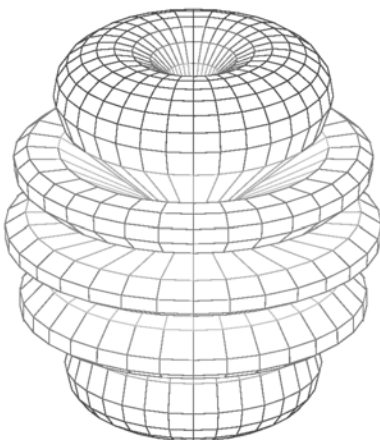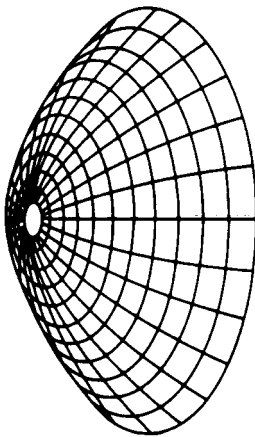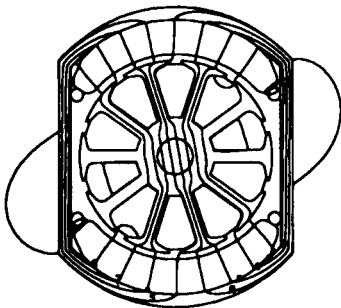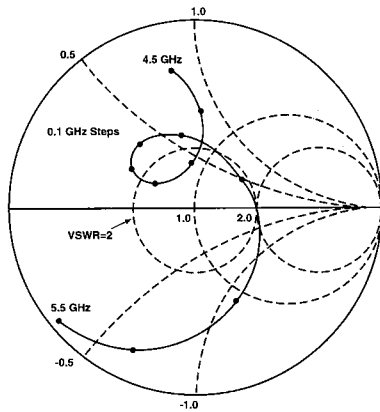
APPLIED COMPUTATIONAL ELECTROMAGNETICS SOCIETY JOURNAL

**Special Issue on
Neural Network Applications in
Electromagnetics**

**Guest Editor
Christos Christodoulou**

July 2003
Vol. 18 No. 2

ISSN 1054-4887

<div align="center">

Special Issue on

Neural Network Applications in Electromagnetics

Guest Editor Introduction

</div>

Neural computing and machine learning algorithms have arrived and are here to stay! In the last ten years neural networks have experienced an unbelievable growth, both in terms of novel neural network architectures that have appeared in the literature, and new applications where neural networks have been used successfully. The high-speed capabilities and "learning" abilities of neural networks can be applied to quickly solving numerous complex optimization problems in electromagnetics, and this special issue shows you how. Even if you have no background in neural networks, the papers that appear in this issue will give you a flavor of the different applications that neural networks can be applied to.

In the past, neural networks (NNs) have been applied to modeling and design of antennas, microstrip circuits, embedded passive components, semiconductor and optical devices, and so on. Today, support vector machines (SVM) have also emerged in the area of machine learning and have been used mainly in the area of pattern recognition and classification. In this issue, two of the papers discuss a machine learning approach to solving electromagnetics problems. One of them compares results between classical neural networks and SVM's.

There are basically four main situations in which NNS and SVMs are good candidates for use in electromagnetics.
1. When closed form solutions do not exist and trial and error methods are the only approaches to solving the problem at hand.
2. When the application requires real-time performance.
3. When faster convergence rates and smaller errors are required in the optimization of large systems.
4. When enough measured data exist to train an NN or an SVM for prediction purposes, especially when no analytical tools exist.
5. When they can be used in conjunction with other numerical techniques for enhancement purposes.

This special issue includes 7 papers all of which are very different yet they have one unifying factor which is the use of NNS and SVM in tackling the problem at hand. The 1st paper is an example of how both neural networks and support vector machines can be used to classify buried objects (a classification problem). The second paper shows how neural networks can be used along with signal processing techniques for bio-medical applications and sensors. In the third paper we see an example of how neural networks can be combined with equivalent circuit formulations and other approaches for modeling of multilayer printed circuits. The fourth paper introduces the use of SVM in training adaptive array antennas for determining the direction of arrival of a signal (DOA). The paper includes a brief introduction of machine learning and support vector machines and how results compare with the more classical existing techniques. The fifth paper demonstrates how measured data can be used to train neural networks to predict

resonances for microstrip antennas at different frequencies and for different dimensions. The sixth paper is an example of how neural networks can be used in problems where no closed-form solutions exist such as the estimation of target orientation using measured radar cross section data. The last paper is a unique example of using neural networks with the asymptotic waveform evaluation (AWE) to speed up the analysis of the method of moments. This combined approach is applied to the solution of a microstrip antenna. Also, several references are included in each paper and the hope is that the reader will be exposed to the wide range of applications that are possible today in the area of electromagnetics using neural networks and machine learning!

Finally, I wanted to thank the following reviewers for helping me with this issue: Chaouki Abdallah, Michael Cryssomallis, Said El-Khamy, K. C. Gupta, Kerim Guney, Nafatli (Tuli) Herscovici, Q. J. Zhang, and Ahmed EL Zooghby. Special thanks go to Atef Elsherbeni for coming up with the idea behind this special issue and being patient and very helpful along the way!

Christos Christodoulou

University of New Mexico

THE APPLIED COMPUTATIONAL ELECTROMAGNETICS SOCIETY

JOURNAL

SPECIAL ISSUE ON
NEURAL NETWORK APPLICATIONS IN ELECTROMAGNETICS

Vol. 18 No. 2   July 2003
TABLE OF CONTENTS

A Comparative Study of NN and SVM-Based Electromagnetic Inverse Scattering Approaches to On-Line Detection of Buried Objects

Salvatore Caorsi[1], Davide Anguita[2], Emanuela Bermani[3], Andrea Boni[3],
Massimo Donelli[3] and Andrea Massa[3]

[1] Dept. of Electronics, University of Pavia, Via Ferrata 1, I-27100 Pavia,Italy
[2] Dept. of Biophysical and Electronic Eng., University of Genoa, Via Opera Pia 11A, I-16145, Genova,Italy
[3] Dept. of Information and Communication Tech.,University of Trento, Via Sommarive 14, I-38050 Trento,Italy

Abstract—**Microwave-based sensing techniques constitute an important tool for the detection of buried targets. In this framework, a key issue is represented by real-time scatterer localization. As far as such a topic is concerned, this paper presents a comparative evaluation of the performances provided by a conventional NN-based inverse scattering technique and by a new SVM-based electromagnetic approach. In order to estimate the effectiveness values of the two methods, realistic configurations and noisy environments are considered and current capabilities, as well as potential limitations, are pointed out. Finally, possible future research work is outlined.**

I. INTRODUCTION

The detection of buried objects by means of interrogating electromagnetic waves is usually a very difficult task. The addressed problem is nonlinear, due to the relation between unknown quantities (object parameters and field distribution) and problem data, it is ill-posed and, generally, only aspect-limited measures are available.

In the past few years, considerable efforts have been devoted to dealing with detection or, more generally, reconstruction problems, and several approaches have been proposed. As far as weak scatterers are concerned, linearized procedures have been applied (see [1], [2], [3] and references cited therein). The use of closed forms of the scattering equations (based on the diffraction theorem) and of the Fast Fourier Transform (FFT) has made it possible to obtain faster processing rates and real-time imaging. Moreover, numerical procedures based on higher-order Born approximations have demonstrated their effectiveness [4], [5].

On the contrary, nonlinear algorithms must be used when strong scatterers are considered. The retrieval problem is usually recast into an optimization one and is effectively solved with minimization techniques [6]-[10]. Unfortunately, large computational resources and a high computational load are needed, thus making

these techniques impracticable (particularly when serial implementations are realized) if real-time performances are required.

In order to speed up the detection process, a key point is the reduction in the number of unknowns. Toward this end, *a-priori* information (if available) on the scenario under test can be very useful. This concept has been successfully exploited in inverse methodologies based on artificial neural networks (NNs) (see [11] (pp. 475-479) and references cited therein). As far as detection problems are concerned, methods based on both multilayered-perceptron [12], [13] and radial-basis-function [14] neural networks have demonstrated their capabilities for on-line retrieving of buried cylindrical scatterers.

Though NN-based approaches have generally offered good performances in solving inverse-scattering problems, they still suffer from several drawbacks not completely solved up to now. From the inductive-theory point of view, the main drawback is the difficult control of the complexity of underlying NN models. By the term *complexity* it is usually meant the capacity of a learning machine to fit the input data. Briefly, if a machine is too complex, it will typically overfit the data, thus losing the property of generalization for new measures not included in the training set. If complexity is too low, the machine will fail to correctly interpret the underlying relations among training samples. The complexity of a learning machine depends on many factors. In the case of NNs, the numbers of hidden layers and neurons, the number of interconnections, and the learning algorithm used for the training process [15] are the predominant parameters. Unfortunately, NNs lack an effective theory suggesting the most suitable NN topologies and/or calibration parameters. An NN adapts its internal parameters (i.e., the weights) automatically in order to best approximate the available training data, but the topology, the transfer function and the other parameters are heuristically selected. At present, there is no good way to determine how many hidden layers or how many hidden nodes each layer

should contain, given the sample data with which to train the NN for the solution of a given problem. From the computational and optimization points of view, NNs exhibit drawbacks as well. The learning process of an NN consists in solving a nonlinear optimization process. Therefore, any conventional optimization algorithm, including the widely used back-propagation procedure, leads to reach a solution that corresponds to one of the local minima of the target cost function. An empirical way to face such a problem is to train several NNs with different starting points, thus overloading the optimization process. However, this choice might result in inability to unambiguously evaluate statistical and systematic errors on neural computations.

A possibility to overcome these drawbacks is based on new results in Statistical Learning Theory (SLT) [16] which lead to new algorithmic paradigms and new computational architectures that, though still based on the NN model, entirely relinquish their biological plausibility to achieve a firm theoretical background. SLT allows one to derive statistical and algorithmic properties that can limit or avoid altogether the NN problems previously described. One of the main contributions in this field has been provided by Vladimir Vapnik [16], who has formulated and formalized the inductive rules that regulate the learning process by example principles. On the basis of these fundamentals, a new learning paradigm, called Support Vector Machine (SVM), has been developed. After initial studies, SVMs are now successfully applied in several fields ranging from pattern recognition to function approximation tasks. From a theoretical point of view, SVMs turn out to be very appealing, as compared with conventional NNs, for the following two basic reasons:

- the constrained-quadratic structure of the optimization problem solved for the learning process;
- the solid statistical theory on which SVMs are based.

In this paper, the detection of buried objects by means of interrogating electromagnetic waves is addressed by using an inductive approach. Within the framework of electromagnetic retrieval, the effectiveness and limitations of the SVM-based strategy are evaluated and a comparative study versus conventional NN-based methods is made. Finally, selected numerical results on realistic configurations and noisy environments are reported and discussed.

II. MATHEMATICAL FORMULATION

Let us consider the problem of determining the unknown parameters of an object buried in a homogeneous soil. With reference to a two-dimensional geometry, let D_S be a lossy region with complex contrast, $\tau_S = [\varepsilon_S - 1] - j\frac{\sigma_S}{2\pi f \varepsilon_0}$, enclosing a circular cylindrical scatterer of radius ρ_B. The dielectric properties of the
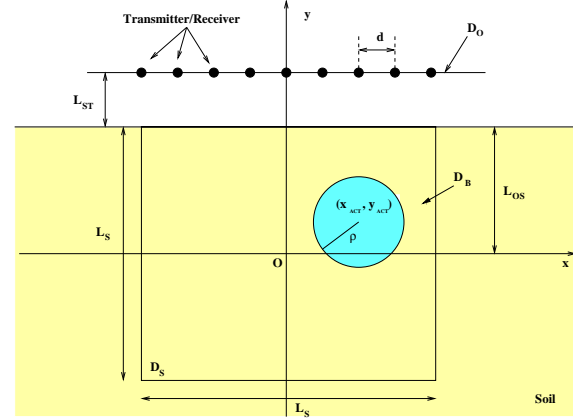
Fig. 1. Geometry of the problem

object are homogeneous, τ_B, and the dielectric profile of the geometry under test (Fig. 1) can be described as follows:

$$\tau(x,y) = \begin{cases} \tau_0 & if\ y > L_{0S} \\ \tau_B & if\ \begin{cases} 0 \le x \le x_B + \rho_B \cos\theta \\ 0 \le y \le y_B + \rho_B \sin\theta \\ 0 \le \theta < 2\pi \end{cases} \\ \tau_S & otherwise \end{cases}$$

(1)

This scenario is illuminated by multiple transmitters lying on D_0 and located, in the upper half space, at the positions $\{(x_t, y_t);\ t = 1, ..., T\}$. The probing fields, $E^{irr}(x,y)$, are radiated in the free space and at a fixed frequency by a known distribution of current filaments parallel to the z-axis. The same probes work as receivers for the anomalous field.

Under these hypotheses, the addressed inverse scattering problem can be mathematically stated as follows. Starting from the knowledge of the anomalous field, E^{tot}, collected at the receiver positions $\{(x_r, y_r);\ r = 1, ..., R\}$

$$E^{tot}(x_r, y_r | x_t, y_t) = E^{inc}(x_r, y_r | x_t, y_t) + \\ + k^2 \int_{D_S} E_S(x, y | x_t, y_t) G_S(x_r, y_r; x, y) \tau(x, y)\, dxdy$$

(2)

determine the set of unknown parameters $\{(x_B, y_B), \rho_B, \tau_B\}$ defining the scatterer under test. In eq. (2):

- $E^{inc}(x_r, y_r | x_t, y_t) = E^{irr}(x_r, y_r | x_t, y_t) + E^{ref}(x_r, y_r | x_t, y_t)$ is the electric field at the receivers in the absence of the object;
- $E^{ref}(x_r, y_r | x_t, y_t)$ is the electric field reflected by the planar interface at the receivers;
- $E_S(x, y | x_t, y_t)$ is the electric field induced inside the reconstruction domain D_S when it contains the object;
- $G_S(x_r, y_r; x, y)$ is the Sommerfeld-Green function for the half-space geometry [6].

Then the solution of the addressed inverse scattering problem requires the determination of the nonlinear function, Φ, defined as follows

$$\underline{\chi} = \Phi\left\{\underline{E}^{tot}\right\} \tag{3}$$

where $\underline{\chi}$ is the "scatterer array" ($\underline{\chi} = [\chi_p; p = 1, ..., P]$ $= [(x_B, y_B), \rho_B, \tau_B]$, P being the number of unknown parameters) and $\underline{E}^{tot}$ is the data array defined as $\underline{E}^{tot} = [E^{tot}(x_r, y_r | x_t, y_t);$ $r = 1, ..., R; t = 1, ..., T]$. This is a *regression problem* in which the unknown function (Φ) must be approximated by the knowledge of a number of known input-output pairs of vectors $\left\{\left(\underline{\chi}\right)_n, \left(\underline{E}^{tot}\right)_n\right\}; n = 1, ..., N$.

III. LEARNING-BY-EXAMPLES STRATEGIES FOR INVERSE SCATTERING PROBLEMS

The inverse-scattering problem described in Section II can be addressed in several ways. From a mathematical point of view, the key issue is to find an approximation, $\hat{\Phi}$, for the unknown function Φ on the basis of a set of samples $\{(\underline{\nu}_n, \underline{e}_n); n = 1, ..., N\}$, $\underline{\nu}_n$ and $\underline{e}_n$ being an input pattern (i.e., a data array $\underline{\nu}_n \equiv \underline{E}^{tot}$) and the corresponding target (i.e., a scatterer array $\underline{e}_n \equiv \left(\underline{\chi}\right)_n$), respectively. This is a typical *learning-by-examples* problem, which is usually faced in the presence of unknown systems with measurable input/output signals. In the following, two approaches based on a multilayer perceptron (MLP) neural network and on an SVM, respectively, will be presented.

A. MLP-NN Approach

Neural networks are distributed computational systems characterized by a multi-layered structure of neurons fully interconnected by weighted links. MLP-NNs can be considered as universal approximators for any function $\Phi : \Re^{R \times T} \to \Re^P$ [17]. Therefore, they are suitable for facing with regression problems characterized by complex nonlinear relations between data and unknowns, such as inverse scattering or buried-object detection problems. In this context, $\Re^{R \times T}$ is the space of arrays representing measurement data, and $\Re^P$ is the space of unknown parameters describing a buried object.

MLP-NN theory [11] suggests approximating Φ by a nonlinear function of the weighted measurement data

$$\widehat{\Phi}\left(\underline{\nu}\right) = \Psi\left\{\underline{\xi}_{L-1}\underline{\underline{w}}^{(L-1, L)} + \underline{b}_L\right\} \tag{4}$$

where L is the number of layers; $\underline{\xi}_l = \Psi\left\{\underline{\xi}_{l-1}\underline{\underline{w}}^{(l-1, l)} + \underline{b}_l\right\}$, $l = 1, ..., L$ being $\underline{\xi}_0 = \underline{\nu}$; Ψ is the nonlinear activation function (e.g., a sigmoid); $\underline{\underline{w}}^{(l-1, l)}$ and $\underline{b}_l$ are the weight matrix and the bias array of the l-th layer, respectively. Given known input-output pairs of vectors (called *training set*),

$\Gamma_{training} = \{(\underline{\nu}_n, \underline{e}_n); n = 1, ..., N\}$, and according to a backpropagation algorithm, weights and biases are computed by minimizing the error function ψ

$$\psi\left\{\underline{\underline{w}}^{(l-1, l)}, \underline{b}_l; l = 1, ..., L\right\} = \sum_{n=1}^{N}\left\|\underline{e}_n - \widehat{\Phi}\left(\underline{\nu}_n\right)\right\| \tag{5}$$

by a gradient descent procedure.

Therefore, the direct solution of the inverse-scattering problem is avoided, and real-time (after the training phase) solutions to buried-object localization are obtained [13]. However, as the error function (5) is non-convex, one of the main drawbacks of the approach is the presence of local minima where the optimization algorithm might be trapped and the solution of which would have no physical significance.

B. SVM-Based Approach

In order to avoid the drawbacks of the NN-based inverse scattering method related to the nonlinear fitting of the training samples, an SVM-based approach is presented. The underlying idea of the SVM procedure is to split the approximation for the nonlinear function Φ into two steps. Instead of performing a nonlinear fitting in the input space, a nonlinear mapping of the training samples from the input space into a larger (possibly infinite) space (i.e, the *feature space*, $\Re^{\tilde{T}}$) is first performed. Then, a simple linear fitting is carried out in the new space, thus avoiding typical nonlinear-fitting drawbacks and keeping the advantages of a linear approach. Moreover, by exploiting some mathematical properties of nonlinear mappings, the evaluation of the data in the feature space is not required, as the SVM does not have to explicitly work in this space.

In more detail, each data array $\underline{\nu}_n$ is mapped into the feature space through a nonlinear transformation $\underline{\varphi} : \Re^{R \times T} \to \Re^{\tilde{T}}$ with $\tilde{T} \gg R \times T$. Then, the samples in the feature space are linearly interpolated by defining a hyperplane with a normal vector $\underline{w}$. Thus, the approximating function is given by

$$\hat{\Phi}\left(\underline{\nu}\right) = \underline{w} \cdot \underline{\varphi}\left(\underline{\nu}\right) + b \tag{6}$$

Among all possible hyperplanes, SVMs find the one that corresponds to a function $\hat{\Phi}$ having at most a deviation ϵ from each target $e_n^{(p)}$ (*), for all the measures $\underline{\nu}_n$, and that, at the same time, is as "flat" as possible. As it is impossible for all the points to lie inside the ϵ band, some errors (ξ_n, ξ_n^*, also called *slack variables*) are allowed and linearly weighted. Mathematically, this description leads to a constrained

---

(*) As up to now it has been possible to synthesize only single-output SVM, we refer to the estimation of a single scatterer array component $e_n^{(p)} = (\chi_p)_n; p = 1, ..., P$.

quadratic optimization problem (CQP) where the regularized cost function γ

$$\gamma\left\{\underline{w},b\right\} = \left\{ \frac{1}{2} \|\underline{w}\|^2 + C \sum_{n=1}^{N} (\xi_n + \xi_n^*) \right\} \quad (7)$$

is minimized over $\underline{w}$ and b, subject to the following constraints:

$$\begin{cases} e_n^{(p)} - \underline{w} \cdot \varphi\left(\underline{\nu}\right) - b \le \epsilon + \xi_n \\ \underline{w} \cdot \varphi\left(\underline{\nu}\right) + b - e_n^{(p)} \le \epsilon + \xi_n^* \quad \forall\, n \\ \xi_n, \xi_n^* \ge 0 \end{cases} \quad (8)$$

The function γ is composed of two terms. The first forces the hyperplane to be as flat as possible, and the second penalizes the deviation of each target from the function $\hat{\Phi}$. The constant C measures the tradeoff between the two terms. It can be shown that this approach can be used to control the complexity of the learning machine, according to the Structural Risk Minimization principle [16]. This principle guarantees a considerable generalization ability of the model, and provides upper bounds to such ability, albeit in a statistical framework. It is also interesting to note that this formulation, which derives from SLT, resembles closely the regularization approach that is usually exploited when dealing with ill-posed problems, like inverse ones [18].

The problem defined by eqs. (7)-(8) is then rewritten in dual form by using the Lagrange multiplier theory. By introducing $2N$ Lagrange multipliers, $\alpha_n, \alpha_n^*, n = 1, \ldots N$, a dual functional, γ_d, to be maximized is obtained (see [19] or [16] for more mathematical details):

$$\gamma_d\left\{\underline{\alpha},\underline{\alpha}^*\right\} = \left\{ -\frac{1}{2} \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\, \varphi\left(\underline{\nu}_i\right) \cdot \varphi\left(\underline{\nu}_j\right) + \right. \\ \left. -\epsilon \sum_{n=1}^{N} (\alpha_n + \alpha_n^*) + \sum_{n=1}^{N} e_n^{(p)}(\alpha_n - \alpha_n^*) \right\} \quad (9)$$

subject to

$$\sum_{n=1}^{N} (\alpha_n - \alpha_n^*) = 0 \quad \alpha_n, \alpha_n^* \in [0, C] \quad (10)$$

as

$$\underline{w} = \sum_{n=1}^{N} (\alpha_n - \alpha_n^*)\, \varphi\left(\underline{\nu}_n\right) \quad (11)$$

Consequently, $\hat{\Phi}$ is equal to

$$\hat{\Phi}\left(\underline{\nu}\right) = \sum_{n=1}^{N} (\alpha_n - \alpha_n^*)\, \varphi\left(\underline{\nu}_n\right) \cdot \varphi\left(\underline{\nu}\right) + b \quad (12)$$

where only the inner product of the nonlinear mapping function (and not the function itself) appears. This is the well-known *kernel trick* that allows one to deal implicitly with nonlinear mappings through the use of Kernel functions

$$k\left(\underline{\nu}_i, \underline{\nu}_j\right) = \varphi\left(\underline{\nu}_i\right) \cdot \varphi\left(\underline{\nu}_j\right) \quad (13)$$

The theory of kernels, that is, the conditions under which equation (13) holds, has been known since the beginning of the last century; it is based on Mercer's theorem [16] and has been applied to pattern recognition tasks since the '60s [20]. However, only recently has the connection with learning machines been well formalized [18]. Kernel functions are positive semidefinite functionals. Therefore, using this property and the fact that the constraints of the above optimization problem are "affine", any local minimum is also a global one, and algorithms exist by which the solution can be found in a finite number of steps [21]. Furthermore, if the kernel is strictly positive definite (that is always the case, except in pathological situations), the solution is also unique. These properties overcome many typical drawbacks of classical neural-network approaches, such as the determination of a suitable minimum, the choice of the starting point, the optimal stopping criteria, and so on.

Since the publication of early seminal works on kernel functions, many functionals have been found that satisfy Mercer's theorem. As far as inverse-scattering problems are concerned, a Gaussian kernel

$$k\left(\underline{\nu}_i, \underline{\nu}_j\right) = exp\left\{ -\left[\frac{\|\underline{\nu}_i - \underline{\nu}_j\|^2}{2\sigma^2} \right] \right\} \quad (14)$$

performing a mapping in an infinite-dimensional feature space [18] and preliminarily used in [19], has demonstrated its effectiveness.

Concerning the SVM parameters, the threshold b is computed by means of the Karush-Khun-Tucker conditions of the CQP at optimality [19], and the hyper-parameters of the problem (σ, q, C and ϵ) are deduced by accomplishing the model-selection task proposed in [22].

Finally, the CQP is solved by a standard optimization algorithm, namely, Platt's SMO algorithm for regression [23].

IV. NUMERICAL RESULTS

In this work, a comparative study of NN and SVM-based approaches is made concerning the localization of a scatterer buried in the soil. Let us consider a square investigation domain $L_S = \lambda$-sided (λ being the free-space wavelength) completely embedded in the ground. The relative permittivity of the soil is $\varepsilon_S = 20.0$ and the conductivity is $\sigma_S = 0.01 \frac{S}{m}$. The center of the region under test is $L_{OS} = \frac{7}{12}\lambda$ deep. The buried object is a lossless circular cylinder of radius $\rho_B = \frac{1}{12}\lambda$ and the relative permittivity of the ground is equal to $\varepsilon_B = 5.0$. This scenario is illuminated by
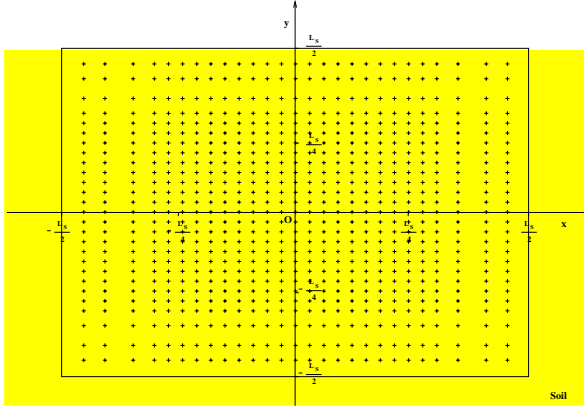
Fig. 2. Training set. Geometrical arrangement of the center of the cylinder under test
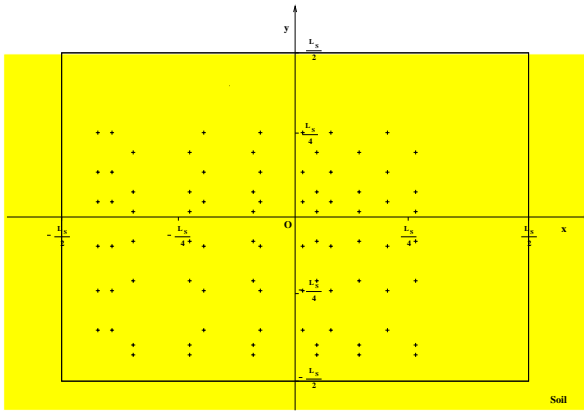
Fig. 3. Test set. Geometrical arrangement of the center of the cylinder under test

an electric line source, located in the upper region with the coordinates $x_t = 0$ and $y_t = \frac{2}{3}\lambda$, $t = T = 1$, and parallel to the air-soil interface. The anomalous field is collected at $R = 16$ measurement points equally spaced ($d = \frac{\lambda}{16}$) and located on a line placed close to the air-soil interface ($L_{ST} = \frac{\lambda}{12}$).

The data used to generate the training set and those used to test the SVM approach, as well as the MLP neural network, were obtained synthetically by a Finite Element code and a PML technique [24]. During the learning phase, the training set, $\{\Gamma_{training}, ; N = 729\}$, was obtained by moving the center of the cylinder inside D_S among the positions shown in Figure 2 and collecting the anomalous field at the receiver positions. As far as the test phase is concerned, $M = 84$ randomly chosen locations of the scatterer (Fig. 3) were considered in order to define the test set $\Gamma_{test} = \{(\underline{v}_m, \underline{e}_m); m = 1, ..., M\}$. An additive Gaussian noise, characterized by the signal-to-noise ratio (SNR)

$$SNR = 10log_{10}\frac{\sum_{r=1}^{T}\sum_{t=1}^{T}|E^{tot}(x_r, y_r|x_t, y_t)|^2}{2T^2\kappa_{noise}^2}$$

(15)

κ_{noise}^2 being the variance of noise, affected the measured signals.

A two-layer MLP-NN [12], characterized by 32 inputs, 32 hidden neurons, and 2 output neurons, was trained by using a delta-bar-delta back algorithm [25] in order to overcome the shortcomings of the gradient-descent procedure and to increase the convergence rate of the standard back-propagation learning algorithm.

Concerning the SVM-based approach, two SVMs were used to estimate the center coordinates of the buried object. Moreover, after the optimal selection procedure, the values of the SVM hyperparameters turned out to be constant quantities equal to $(C)_{x_B} = (C)_{y_B} = 100$ and $\epsilon = 0.001$. On the contrary, the variance values of the Gaussian kernels, $(\sigma^2)_{x_B}$ and $(\sigma^2)_{y_B}$, were determined independently of each scenario under test.

A. Definitions

In order to quantitatively estimate the effectiveness of the presented approaches, some error values are defined. Let us introduce the

- *local errors* on the center location, δ_x^u and δ_y^v:

$$\begin{aligned} \delta_x^u &= \frac{|x_{act}^u - x_{rec}^{v(u)}|}{d_{max}} & u = 1, ..., U; \\ & & v(u) = 1, ..., V(u) \\ \delta_y^v &= \frac{|y_{act}^v - y_{rec}^{u(v)}|}{d_{max}} & v = 1, ..., V; \\ & & u(v) = 1, ..., U(v) \end{aligned}$$

(16)

- *local average errors* on the object localization, $\zeta_x = \{\zeta_x^u, u = 1, ..., U\}$ and $\zeta_y = \{\zeta_y^v, v = 1, ..., V\}$:

$$\begin{aligned} \zeta_x^u &= \frac{|x_{act}^u - \frac{1}{V(u)}\sum_{v(u)=1}^{V(u)} x_{rec}^{v(u)}|}{d_{max}} & u = 1, ..., U \\ \zeta_y^v &= \frac{|y_{act}^v - \frac{1}{U(v)}\sum_{u(v)=1}^{U(v)} y_{rec}^{u(v)}|}{d_{max}} & v = 1, ..., V \end{aligned}$$

(17)

- *global average errors*, Θ_x and Θ_y:

$$\begin{aligned} \Theta_x &= \frac{1}{d_{max}}\sqrt{\frac{1}{U}\sum_{u=1}^{U}\left[x_{act}^u - \frac{\sum_{v(u)=1}^{V(u)} x_{rec}^{v(u)}}{V(u)}\right]^2} \\ \Theta_y &= \frac{1}{d_{max}}\sqrt{\frac{1}{V}\sum_{v=1}^{V}\left[y_{act}^v - \frac{\sum_{u(v)=1}^{U(v)} y_{rec}^{u(v)}}{U(v)}\right]^2} \end{aligned}$$

(18)

where the subscripts rec and act refer to the estimated and actual coordinates of the scatterer, respectively; $d_{max} = L_S$ is the maximum error in defining the coordinates of the center of the circular scatterer when it is contained in the investigation domain, D_S.
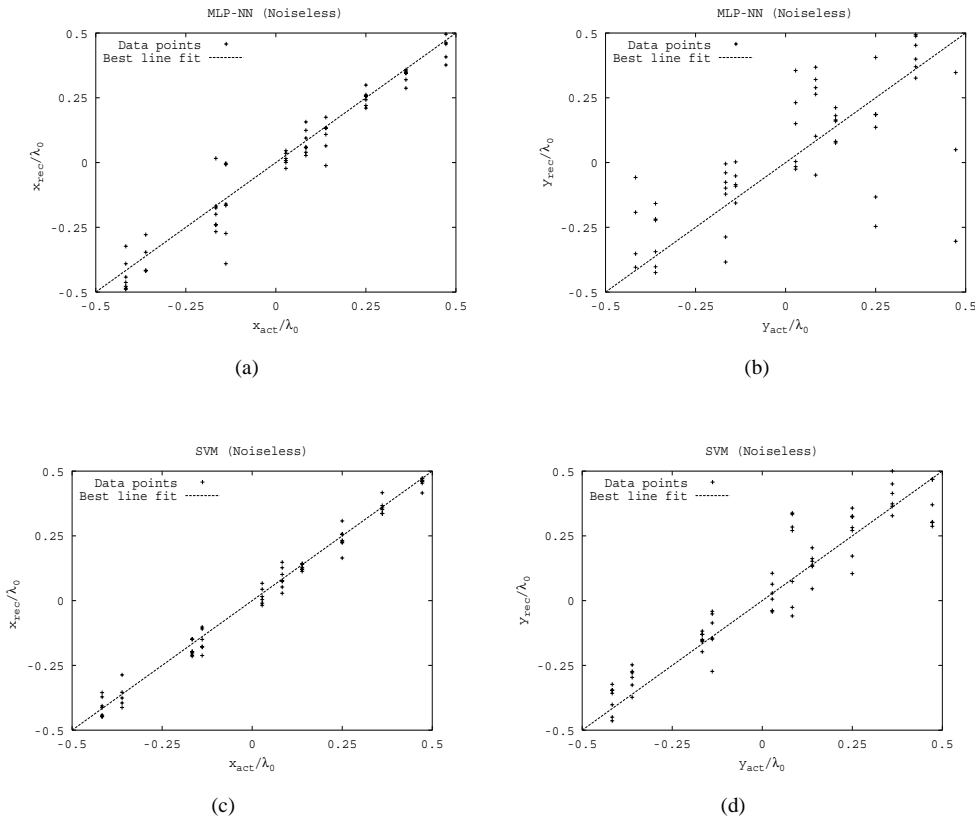
Fig. 4. Reconstructed data versus actual data (Noiseless Case). (a)-(b) MLP-NN approach, (c)-(d) SVM-based approach. (a)-(c) $\frac{x_B}{\lambda_0}$ and (b)-(d) $\frac{y_B}{\lambda_0}$

B. Numerical Assessment - Scenario 1

The first example is aimed at evaluating the possibility of locating the buried object starting from the knowledge of the measured electric field and assuming the knowledge of the soil characteristics to be a-priori information about the geometry under test. Consequently, the incident field is a known quantity and the data array is defined as follows: $\underline{E^{scatt}} = \left[E^{tot}(x_r, y_r | x_t, y_t) - E^{inc}(x_r, y_r | x_t, y_t) \right]; r = 1, ..., R; t = 1, ..., T \right]$. In this context, the SVM parameters have been chosen equal to $\left(\sigma^2 \right)_{x_B} = 0.64$ and $\left(\sigma^2 \right)_{y_B} = 0.32$.

Figure 4 shows the localization results for the examples making up the test set and obtained by using the MLP-NN (Fig. 4(a)-(b)) and the SVM-based procedure (Fig. 4(c)-(d)), respectively. As can be observed, a good accuracy concerning the center location is achieved along the two reference axes and for both the MLP-NN and SVM-based approaches. In particular, even if the detection accuracy decreases as the distance from the air-soil interface increases, good localizations are achieved in the whole domain, as confirmed by the statistics shown in Table I. In particular, as far as the scatterer depth estimation is concerned, the SVM

sharply reduces the error of the MLP-NN, reaching an average error equal to $< \delta_y^v >_{SVM} = 0.0584$ ($< \delta_y^v >_{MLP} = 0.1004$ being the average error made by the MLP-NN approach). Moreover, it should be noted that the time required for the SVM training is about one tenth of the one required by the MLP-NN, whereas there is no significant difference between the computation times taken by the two approaches for the object localization (i.e, after the learning phase).

TABLE I

SCENARIO 1 (*Noiseless Case*). LOCAL ERROR STATISTICS

	$< \delta_x^u >$	$\max\{\delta_x^u\}$	$\min\{\delta_x^u\}$
MLP	0.0347	0.2098	$7.14 \cdot 10^{-4}$
SVM	0.0177	0.1243	$7.30 \cdot 10^{-5}$

	$< \delta_y^v >$	$\max\{\delta_y^v\}$	$\min\{\delta_y^v\}$
MLP	0.1004	0.5631	$5.31 \cdot 10^{-3}$
SVM	0.0584	0.3487	$7.44 \cdot 10^{-4}$

In order to assess the robustness of the learning-based retrieval strategies, a noisy environment has been taken into account. For all the simulations, the buried cylinder and the electromagnetic scenario are unchanged and characterized by the same dielectric
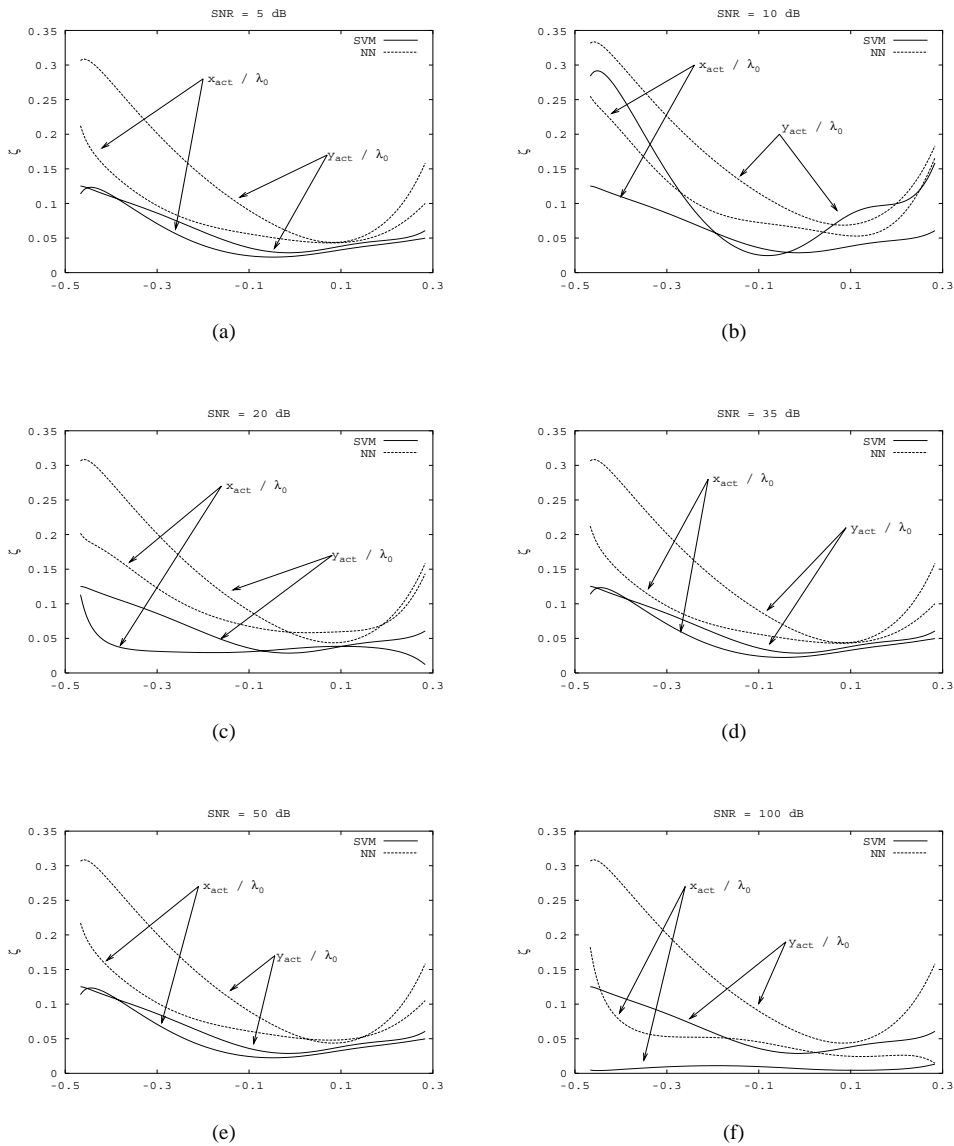
Fig. 5. Scenario 1 (*Standard Validation*). Local average errors of the MLP-NN and SVM-based procedures for different signal-to-noise ratios: (*a*) $SNR = 5\,dB$, (*b*) $SNR = 10\,dB$, (*c*) $SNR = 20\,dB$, (*d*) $SNR = 35\,dB$, (*e*) $SNR = 50\,dB$, and (*f*) $SNR = 100\,dB$

properties as during the training phase. The local average errors are given in Figure 5. For different signal-to-noise ratios, the plots of ζ_x and ζ_y related to both the MLP-NN and the SVM-based procedures are shown. As expected, the estimation of the scatterer depth turns out to be more difficult than the horizontal detection. However, the performances guaranteed by the SVM procedure generally outperform those achieved by the MLP-NN approach. Concerning the dependence of the reconstruction properties on the SNR value, the scatterer is located quite correctly, and $\zeta_x \leq 0.025$ whatever the noisy case considered. Moreover, ζ_y is greater than 0.05 only in the interface regions (i.e., near the air-soil interface and at the bottom of the investigation area). This behavior is not surprising, as confirmed by the experimental results reported in [26], where the problem of the pollution of the useful signal due to the reflections of the air-ground interface is clearly pointed out.

Another evaluation of the robustness of the proposed approaches has also been obtained by carrying out the so-called *cross validation test*. The two methods have been trained with a noisy data set (i.e., a data set whose samples are related to an assigned signal-to-noise ratio $SNR_{Training}$) and tested with a test set computed in a different noisy environment (SNR_{Test}).

(a)

(b)

(c)

(d)
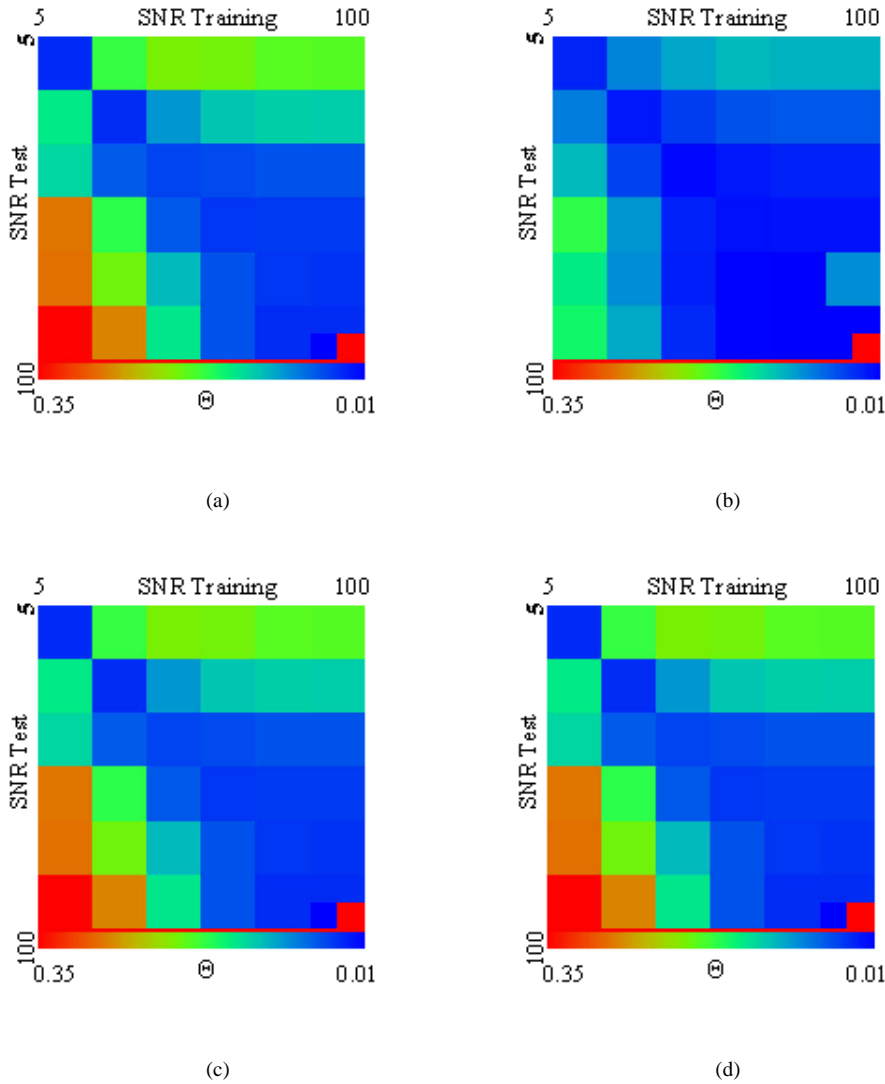
Fig. 6.   Scenario 1 (*Cross Validation*). Global average errors: (*a*) $\{\Theta_x\}_{MLP-NN}$, (*b*) $\{\Theta_x\}_{SVM}$, (*c*) $\{\Theta_y\}_{MLP-NN}$, and (*d*) $\{\Theta_y\}_{SVM}$

Figure 6 shows a color-level representation (*) of the global average errors for different values of signal-to-noise ratio of the training and test sets ranging from $5\,dB$ to $100\,dB$. Figures 6(*a*)-(*c*) and 6(*b*)-(*d*) refer to the MLP-NN approach and the SVM method, respectively. As expected, the smallest values of the global errors are reached when the same noisy environment is considered for both the training and test data-sets. Otherwise, the SVM method always outperforms the MLP-NN approach in the estimation of the horizontal coordinate of the scatterer (Θ_x). As far as the depth of the scatterer location is concerned, similar conclusions can be drawn for the region defined by the following ranges: $SNR_{Training} \geq 10\,dB$ and

---

* The two pixels at the right-bottom angles of the plots indicate the minimum and maximum values of the global errors.

$SNR_{Test} \geq 10\,dB$. Otherwise, the comparative study does not provide any significant information.

C. Numerical Assessment - Scenario 2

In the second example, a more complex scenario has been preliminarily considered. No information about the soil is available and the problem data are the measures of the anomalous field, $\underline{E}^{tot} = [E^{tot}(x_r, y_r | x_t, y_t); \quad r = 1, ..., R; \; t = 1, ..., T]$. As far as the choice of the hyperparameters is concerned, the same value equal to 0.04 has been assumed for $(\sigma^2)_{x_B}$ and for $(\sigma^2)_{y_B}$.

As expected (Fig. 7), the performances of the learning-by-examples strategies considerably reduce, as compared with Scenario 1. However, the higher effectiveness of the SVM-based procedure than that

Fig. 7. Scenario 2 (*Standard Validation*). Local average errors of the MLP-NN and SVM-based procedures for different signal-to-noise ratios in the range between $SNR = 5\,dB$ and $SNR = 100\,dB$

of the MLP-NN method is confirmed. Starting from $SNR = 20\,dB$, the local error values turn out to be smaller than 0.15. On the contrary, the performances of the MLP-NN method strongly worsen, as indicated by the dashed lines in Figure 7.

V. CONCLUSIONS

In this paper, two inductive methods for the detection of buried objects have been extensively compared. Starting from an integral formulation of the scattering equations, the buried-object localization has been reformulated as a regression problem and successively solved by means of two learning-by-examples strategies, namely, the MLP-NN approach and the SVM-based procedure. The estimation of the effectiveness

of the proposed procedures has been carried out in different test cases that have clearly confirmed the higher robustness of the SVM-based approach in solving difficult approximation problems as compared with traditional neural networks. Several scenarios have been considered and the behaviors of the two inductive models have been illustrated for different operating conditions. The obtained results have demonstrated the successful application of the SVM-based procedure to solve inverse-scattering problems. Future work, currently under development, will be devoted to improving the performances of the SVM-based procedure and to determining customized kernel functions.

REFERENCES

[1] C. Dourthe, C. Pichot, J. C. Dauvignac, and J. Cariou, "Inversion algorithm and measurement system for microwave tomography of buried object," *Radio Science*, vol. 35, pp. 1097-1108, 2000.

[2] T. J. Cui and W. C. Chew, "Novel diffraction tomographic algorithm for imaging of two-dimensional targets buried under a lossy earth," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 38, pp. 2033-2041, 2000.

[3] A. Joisel and J. C. Bolomey, "Rapid microwave imaging of living tissues," *Proc. of the SPIE - The International Society for Optical Eng.*, vol. 39, pp. 320-330, 2000.

[4] P. Chaturvedi and R. G. Plumb, "Electromagnetic imaging of underground targets using constrained optimization," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 33, pp. 551-561, 1995.

[5] T. J. Cui, W. C. Chew, A. A. Aydiner, and S. Chen, "Inverse scattering of two-dimensional dielectric objects buried in a lossy earth using distorted Born iterative method," *IEEE Trans. Geoscience and Remote Sensing*, vol. 39, pp. 339-345, 2001.

[6] S. Caorsi, G. L. Gragnani, and M. Pastorino, "Numerical electromagnetic inverse-scattering solutions for two-dimensional infinite dielectric cylinders buried in a lossy half-space," *IEEE Trans. Microwave Theory and Technique*s, vol. 41, pp. 352-356, 1993.

[7] K. Belkebir, R. E. Kleinman, and C. Pichot, "Microwave imaging-location and shape reconstruction from multifrequency scattering data," *IEEE Trans. on Microwave Theory and Techniques*, vol. 45, pp. 469-76, 1997.

[8] I. T. Rekanos, T. V. Yioultsis, and T. D. Tsiboukis, "Inverse scattering using the finite-element method and a nonlinear optimization technique," *IEEE Trans. on Microwave Theory and Techniques*, vol. 47, pp. 336-344, 1999.

[9] S. Caorsi, A. Massa, M. Pastorino, "A computational technique based on a real-coded genetic algorithm for microwave imaging purposes," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 38, pp. 1697-1708, 2000.

[10] S. Caorsi, A. Massa, and M. Pastorino, " A microwave inverse scattering technique for image reconstruction based on a genetic algorithm," *IEEE Trans. on Instrumentation and Measurement*, vol. 49, pp. 573-578, 2000.

[11] C. Christodoulou and M. Georgiopoulos. *Applications of Neural Networks in Electromagnetics*. Boston, Artech House, 2001.

[12] S. Caorsi and P. Gamba, "Electromagnetic detection of dielectric cylinders by a neural network approach," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 37, pp. 820-827, 1999.

[13] E. Bermani, S. Caorsi, and M. Raffetto, "An inverse scattering approach based on a neural network technique for the detection of dielectric cylinders buried in a lossy half-space," in *Progress in Electromagnetic Research*, vol. 26, pp. 69-90, 2000.

[14] I. T. Rekanos, "On-line inverse scattering of conducting cylinders using radial-basis -function neural networks," *Microwave and Optical Technology Letters*, vol. 28, pp. 378-380, 2001.

[15] M. Anthony and P. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge, University Press, 1999.

[16] V. N. Vapnik. *The Nature of Statistical Learning Theory*. John Wiley & Sons, New York, 1999.

[17] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Macmillan, New York, 1994.

[18] B. Schölkopf and A. Smola. *Learning with Kernels*, The MIT Press, 2002.

[19] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge, University Press, 2000.

[20] M. A. Aizerman, E. M. Braverman, and L. I. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition learning," *Automation and Remote Control*, vol. 25, pp. 821-837, 1964.

[21] C.-J. Lin, "Asymptotic convergence of an SMO algorithm without any assumptions," *IEEE Trans. on Neural Networks*, vol. 13, no. 1, pp. 248-250, 2002.

[22] D. Anguita, A. Boni, and S. Ridella, "Evaluating the generalization ability of Support Vector Machines through the bootstrap," *Neural Processing Letters*, vol. 11, pp. 1-8, 1999.

[23] J. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods - Support Vector Learning*, The MIT Press (B. Schölkopf, C. Burges, and A. Smola, Eds.), 1999.

[24] S. Caorsi and M. Raffetto, "Perfectly matched layer for truncation of the finite element meshes in layered half-space geometries and applications to the electromagnetic scattering by buried objects," *Microwave and Optical Tech. Lett.*, vol. 19, pp. 427-434, 1999.

[25] R. A. Jacobs, "Increased rates of convergence through learning rate adaption," *Neural Networks*, vol. 1, no. 4, pp. 295-307, 1988.

[26] P. Millot, P.Borderies, E. Guillanton, E. Le Brusq, J. Y. Dauvignac, and Ch. Pichot, "Radar imaging of shallow buried objects using an ultra-wideband technique with specific antennas and synthetic antenna processing," in *Proc. PIERS 2000*, Cambridge, MA, p. 298, July 2000.

Salvatore Caorsi received the laurea degree in electronic engineering from the University of Genoa, Italy, in 1973. Since 1994, he has been a Full Professor of Electromagnetic Compatibility at the Department of Electronics, University of i Pavia, Italy. He is also teaching the course of Antennas at the University of Genoa. His primary activities focus in applications of electromagnetic field to telecommunications, artificial vision and remote sensing, biology and medicine. In particular, he is working on research project concerning human hazard to electromagnetic exposure, numerical methods for solving electromagnetic i problems, wave interaction in the presence of non linear media, inverse scattering and microwave imaging, and electromagnetic compatibility. Prof. Caorsi is the past president and founding member of the Inter-University Research Center for the Interactions Between Electromagnetic Fields and Biological Systems (ICEmB). Moreover, he is a member of the Elettrotecnica ed Elettronica Italiana (AEI), the European Bioelectromagnetic Association (EBEA), and the European Society for Hyperthermic Oncology (ESHO).

Davide Anguita graduated in Electronic Engineering in 1989 and obtained the Ph.D. in Computer Science and Electronic Engineering at the University of Genova, Italy, in 1993. After working as a research associate at the International Computer Science Institute, Berkeley, USA, on special-purpose processors for neurocomputing, he joined the Dept. of Biophysical and Electronic Engineering at the University of Genova, where he teaches digital electronics. His current research focuses on industrial applications of artificial neural networks and kernel methods and their implementation on digital and analog electronic devices. He is a member of IEEE and chair of the Smart Adaptive Systems committee of the European Network on Intelligent Technologies (EUNITE).

Emanuela Bermani received the "laurea" degree in Electronic Engineering, and the Ph.D. degree in "Electronic and Computer Sciences" from the University of Pavia, Pavia, Italy, in 1997 and in 2000, respectively. She is currently an Assistant Professor in the Department of Information and Communication Technology, University of Trento, Italy. Her main interests are in the fields of electromagnetic direct and inverse scattering, neural networks in electromagnetics, and biomedical applications of electromagnetic fields.

Andrea Massa received the "laurea" degree in Electronic Engineering from the University of Genoa, Genoa, Italy, in 1992 and Ph.D. degree in electronics and computer science from the same university in 1996. From 1997 to 1999 he was an Assistant Professor of Electromagnetic Fields at the Department of Biophysical and Electronic Engineering (University of Genoa) teaching the university course of Electromagnetic Fields 1. Since 2001, he has been Associate Professor of at the University of Trento where he teaches the courses of "Electromagnetic Fields 1", "Electromagnetic Techniques for Biomedical and Industrial Diagnostics", and "Wireless communications 1". At present, Prof. Massa is the director of the ELEDIALab and a member of the Inter-University Research Center for Interactions Between Electromagnetic Fields and Biological Systems (ICEmB). His research work since 1992 has been principally on electromagnetic direct and inverse scattering, optimization techniques for microwave imaging, wave propagation in presence of nonlinear media, applications of electromagnetic fields to telecommunications, medicine and biology.

Andrea Boni was born in Genova, Italy, in 1969 and graduated in Electronic Engineering in 1996. He received a Ph.D. degree in Electronic and Computer Science in 2000. After working as research consultant at DIBE, University of Genova, he joined the Department of Information and Communication Technologies, University of Trento, Italy, where he teaches digital electronics. His main scientific interests are on the study and development of digital circuits for advanced information processing, with particular attention to programmable logic devices, digital signal theory and analysis, statistical signal processing, statistical learning theory and support vector machines. The application of such interests focuses on identification and control of non-linear systems, pattern recognition, time series forecasting, signals processing.

Massimo Donelli received the "laurea" degree in Electronic Engineering from the University of Genoa, Italy, in 1998. He is currently a Ph. D. student in Space Science and Engineering and works as a consultant for the Department of Information and Communication Technology, University of Trento, Italy. His main interests are on electromagnetic inverse scattering, adaptive antennas synthesis, optimization techniques for microwave imaging, wave propagation in superconducting materials.

Neural Network Approaches To The Processing of Experimental Electro-Myographic Data from Non-Invasive Sensors

Francesco Carlo Morabito, Maurizio Campolo
University "Mediterranea" of Reggio Calabria
Faculty of Engineering, DIMET
Reggio Calabria (Italy)
E-mail: morabito@unirc.it, campolo@ing.unirc.it

Abstract

Learning theories and algorithms for both supervised and unsupervised Neural Networks (NNs) have already been accepted as relevant tools to cope with difficult problems based on the processing of experimental electromagnetic data. These kinds of problems are typically formulated as inverse problems. In this paper, in particular, the electrical signals under investigations derive from experimental electromyogram interference patterns measured on human subjects by means of non-invasive sensors (surface ElectroMyoGraphic, sEMG, data). The monitoring and the analysis of dynamic sEMG data reveals important information on muscles activity and can be used to clinicians for both preventing dramatic illness evolution and improving athletes performance. The paper proposes the use of Independent Component Analysis (ICA), an unsupervised learning technique, in order to process raw sEMG data by reducing the typical "cross-talk" effect on the electric interference pattern measured by the surface sensors. The ICA is implemented by means of a multi-layer NN scheme. Since the IC extraction is based on the assumption of stationarity of the involved sEMG recording, which is often inappropriate in the case of biomedical data, we also propose a technique for dealing with non-stationary recordings. The basic tool is the wavelet (time-frequency) decomposition, that allows us to detect and analyse time-varying signals. An auto-associative NN that exploits wavelet coefficients as an input vector is also used as simple detector of non-stationarity based on a measure of reconstruction error. The proposed approach not only yields encouraging results to the problem at hand, but suggests a general approach to solve similar relevant problems in some other experimental applications of electromagnetics.

1. Introduction

Most relevant medical problems are today faced by processing (by visual inspection or some automatic means) electrical signals detected on the human body. Evaluation of patient populations often includes the use of ancillary tests for diagnosis and/or prognosis. Data sets collected from these diagnostic tests, such as the Electroencephalogram (EEG), the Electromyogram (EMG), the Electrocardiogram (ECG) and, more recently, functional Magnetic Resonance Imaging (fMRI), tend to be complex, large and high-dimensional. The trend towards digitization of the traditionally analog

EEG, EMG and ECG signals has coincided with the development of computing power and multivariate signal processing techniques capable of manipulating and analyzing such large data sets [Akay M.,1997].

The use of Independent Component Analysis (ICA), an unsupervised learning technique which generalizes Principal Component Analysis (PCA), commonly implemented through Neural Network (NN) schemes, is proposed in this study to process experimental biomedical data. Applied to sEMG (surface ElectroMyoGraphy) data, ICA results in numerous spatially-independent patterns, each associated with a unique time-course, providing a way to separate different electrical signals coming from different muscle activities [Jung T.P., 2000]. In contrast to the variable nature of the surface EMG recorded from a single muscle in isolation, ICA of the sEMG from several muscles simultaneously allows the detection of highly reproducible components for example in the sEMG of the face and the throat during swallowing and in the sEMG of arm muscles during reaching movements [McKeown M.J., 2002].

The researches reported in the present study show important applications in the study of some neurological diseases, and in the monitoring of athletic activities for improving significantly the potential of athletes as well as the capabilities of normal subjects in daily actions, since it makes it possible, in principle, to enhance motor coordination. Also, musculo-skeletal disorders are the first cause of patient-physician encounters in the industrialized countries [IEEE Engineering in Medicine and Biology, 2001].

This paper is organized as follows. In Section 2 the type of data coming from electrical activity of muscles will be discussed. In Section 3 we shall propose the McKeown idea of motion through integration of sub-movements [McKeown M.J., 200b]. The computational model incorporating sub-movements will be presented in Section 4. Section 5 is devoted to the proposal of NN schemes to implement ICA. Section 6 will report the results achieved by processing the experimental data. The assumption of stationarity of the electrical signals will be relaxed in Section 7, where the wavelet approach will be proposed. Finally, some conclusions are drawn.

2. ElectroMyographic Data

When skeletal muscle fibers contract, they conduct electrical activity (action potentials, APs) that

can be measured by electrodes affixed to the surface of the skin above muscles [Akay M., 1997]. As the APs pass by the electrodes, spikes of electrical activity are observed and pulses of muscle fiber contractions are produced. Small functional groups of muscle fibers, termed motor units (MUs), contract synchronously, resulting in a motor unit action potential (MUAP). To sustain force, an MU is repeatedly activated by the central nervous system several times per second. The repetition, or average, firing rate is often between 5 and 30 times per second (or faster). The electromyographic (EMG) signal is widely used as a suitable means to have access to physiological processes involved in producing joint movements. The information extracted from the EMG signals can be exploited in several different applications. The typical sensors used for EMG are needle (unipolar or bipolar) sensors. The experimental data here analysed come from non-invasive surface EMG sensors, that present the cross-talk effect, i.e., they detect electrical activities from several muscles simultaneously in action.

3. Sensorimotor integration of sub-movements

A growing body of evidence suggests movements which appear smooth to the naked eye are actually composed of the temporal and spatial superposition of discrete sub-movements precisely recruited and coordinated by the central nervous system [Harris C.M., 1998]. However, the spatial and temporal overlap of sub-movements has generally made it impossible, with the common computational tools available to the neuroscientist, to isolate the effects of individual sub-movements [Sejnowski T.J., 1998].

Extensive computational expertise is required to adequately interpret the data gleaned from the experiments. Detection of non-stationarity in the sEMG and kinematic variables is necessary to detect the onset of temporally overlapping sub-movements. We investigate the information-theoretic considerations of channel capacity and bandwidth as important determinants in the selection and sensorimotor integration of individual sub-movements.

4. Computational Models incorporating Sub-movements

Some computational approaches have attempted to model reaching movements as incorporating sub-movements; however, they have not addressed many of the unanswered questions regarding the characteristics of sub-movements. Others have attempted to model reaching movements without considering sub-movements at all. Smoothness, an empirical observation of motor movements, has often used as a cost function to optimise the models. Rather than define sub-movements on the basis of the velocity profiles, in this project the sub-movements are defined on the basis of muscular activity. Empirically, experienced physical therapists describe "efficiency" of motor movements as subjects

progressively recover. At some stage of rehabilitation, people are able to mimic normal kinematics but still complain of muscle aching and fatigue due to excessive muscle co-contraction.

Intuitively, sub-movements are groups of muscles that have the tendency to activate together following a common neural input. We assert that a sub-movements is "hard-wired" by adulthood, in the sense that it may be encoded in the spinal cord as part of a Central Pattern Generator (CPG), and also partly reflect the anatomical distribution across several muscles of a single nerve root exiting the spinal cord. To suggest a computational model of sub-movements, we initially make the stationarity assumption. Since the EMG is an indirect measure of the neural command to the muscle, the Mutual Information (MI) can be used as a metric to infer the recordings from two EMG electrodes contain common neural input. M. McKeown has proposed using ICA for the analysis of sEMGs, demonstrating that the Independent Components (ICs) are more strongly coupled with ongoing brain rhythms (EEG) than the sEMGs recordings of individual muscles [McKeown M.J., 2000a]. The ICA model can be used to provide a useful starting point for the rigorous definition of a sub-movement upon which more elaborate models can be created. Consider numerous simultaneous sEMG recordings deriving from several electrodes distributed over many muscles during a coordinated cortically-controlled movement. If we model the sEMGs recorded from each electrode to be the linear superposition of activity from different group of muscles (possibly encoded with CPGs) that tend to co-activate, the, the goal is to estimate the cortical modulation of the commonly influenced muscles. A single sub-movement is defined as $m(t) = U\ C(t)$, $t=t0 \rightarrow tn$, where m is a column vector, with m_j representing the muscle electrical activity contributing to the jth electrode as a function of time, U is a stationary column vector representing the relative weighting that a given cortical command gives to the different muscle areas, and C(t) is the unknown scalar neural command over time. If several, e.g. p, sub-movements during a complex movement are temporally (and spatially) overlapping, the linear combination of $m_k(t_k)$ outputs M(t), the total muscle electrical activity over the duration of the whole movement and M_j is the electrical activity recorded from the jth electrode, Ck represents the relative activation of the kth sub-movement by an independent cortical command, and the matrix $U_{j,k}$ has as its columns, U_k, the vectors defining the different sub-movements. If we assume that for a given time-period, say T, a constant number of sub-movements, c, are simultaneously active, thus, we have $M = UC$, where M is the matrix of the electrical activity, C is the matrix of presumed independent cortical commands, and U is a matrix defining the sub-movements. The goal is then, given the recordings from the electrodes, and not knowing U, to estimate the different cortical influences, **C**. If the C_k are assumed to be independent, and c can be estimated, this is possible through the ICA.

5. Neural models of ICA

Independent Component Analysis (ICA) can easily be introduced as a straightforward evolution of the well-known statistical technique referred to as Principal Component Analysis (PCA). Nevertheless, it is also possible to investigate the main ideas behind ICA from the perspectives of both learning/neural systems and signal processing (blind source separation). A good definition of ICA can be found in [Lee T.W., 1998]: ICA is a method for finding a linear non-orthogonal co-ordinate system in any multivariate data. The directions of the axes of this co-ordinate system are determined by both the second and higher order statistics of the original data. The goal is to perform a linear transformation which makes the resulting variables as statistically independent from each other as possible. In contrast to correlation-based transformations such as PCA, ICA not only decorrelates the signals, through second-order statistics, but also reduces higher-order statistical dependencies. Blind source separation by ICA has received attention because of its potential applications in signal processing. Here, the goal is to recover independent sources given only sensor observation that are unknown linear mixtures of the latent (unobserved), possibly independent, source signals. In parallel to blind source separation researches, the ICA emerged within the framework of unsupervised learning. In particular, Linsker [Linsker R.] firstly proposed an algorithm based on information theory that was then used to maximize the mutual information between the inputs and the outputs of a NN. Each neuron of an "output" layer should be able to encode features that are as statistically independent as possible from other neurons over another ensemble of "inputs". The statistical independence of the outputs implies that the multivariate probability density function (pdf) of the outputs can be factorised as a product of marginal pdf's. Bell and Sejnowski [Bell A.J., 1995], derived stochastic gradient learning rules for achieving the prescribed maximization. The same Authors put the problem in terms of an information-theoretic framework and demonstrated the separation and deconvolution of linearly mixed sources [Bell A.J., 1996].

Among the various approaches proposed in the literature to implement the ICA, the approach used by McKeown [Lee T.W., 1999] is the algorithm developed by Bell and Sejnowski [Bell A.J., 1995] which is based on an Infomax NN, where a self-organizing algorithm is used to maximize the information transferred in a network of non-linear units. The general framework of ICA is now simply described as the blind separation problem, typically introduced by the "cocktail party problem": we have n different sources s_i (that is, the speakers i=1,...,n) and m different linear mixtures x_j (that is, the microphones j=1,...,m). By referring to $\underline{x}$ as the matrix of the observed signals, and as $\underline{s}$ the matrix of the independent components, the matrix $\underline{\underline{W}}$, called unmixing matrix, satisfies the following property:

$$\underline{\underline{s}} = \underline{\underline{W}} \cdot \underline{\underline{x}} \qquad (1)$$

or, by defining the mixing matrix $\underline{\underline{A}}$ as:

$$\underline{\underline{x}} = \underline{\underline{A}} \cdot \underline{\underline{s}} \qquad (2)$$

then the mixing and unmixing matrixes are related by the following equation:

$$\underline{\underline{W}}^{-1} = \underline{\underline{A}} \qquad (3)$$

5.1 The ICA based on the information maximization by using a neural network approach

Bell and Sejnowski derived a self-organizing learning algorithm to maximize the information transferred to a NN of non-linear units. The non-linear transfer functions pick up the higher-order moments of the statistical distribution of the input data, and, moreover, they are able to reduce the redundancy in the output data. Higher-order methods use information on the distribution of $\underline{x}$ that is not contained in the covariance matrix. This fact becomes meaningful when the distribution of $\underline{x}$ is non Gaussian, since it is possible to assume that the covariance matrix of a zero mean Gaussian variable, contains the whole information carried by this variable. By defining the differential entropy for a continuous random variable x as:

$$H(x) = -\int_{-\infty}^{+\infty} f_x(x) \cdot \ln\left[f_x(x)\right] \cdot dx \qquad (4)$$

when $f_x(x)$ is the probability density function of the considered variable. The conditional differential entropy is defined as follows:

$$H(y|x) = -\int_{-\infty}^{+\infty} f_x(x) \int_{-\infty}^{+\infty} f_y(y|x) \cdot \ln\left[f_y(y|x)\right] \cdot dy \cdot dx \qquad (5)$$

It represents to the variations that occur in the information carried by y when x is observed. Finally the mutual information between two variables x and y is given by:

$$MI(x,y) = H(x) - H(x|y) = H(y) - H(y|x) \qquad (6)$$

This quantity measures the information that is added to x when y is observed, or to y when x is observed. The mutual information of (x, y) zeroes, when and only when the variables are independent. The Bell-Sejnowski approach is based on the use of a NN able to minimize the mutual information between the input $\underline{x}$ and the output $\underline{y}$ of the neural network where $\underline{y}$ are the independent components. If we suppose to have noise-free input data, $\underline{y}$ can be obtained from $\underline{x}$ by a deterministic manner: in this case, $H(\underline{y}|\underline{x})$ assumes its lowest value ($-\infty$). The problem in this case is that the density functions of the unknown components cannot be computed, and therefore the $H(\underline{y}|\underline{x})$ is difficult to be estimated. This drawback can be overcame by taking into account that, if $\underline{y}$ can be computed from $\underline{x}$ by an invertible continuous deterministic mapping, the maximization of $MI(\underline{x}|\underline{y})$ corresponds to maximize the entropy of the outputs. In the NN case, we have to maximize the $H(\underline{y})$ with respect to the network parameters $\underline{w}$. If we have just one input x and one output y, if the mapping from x to y is defined as y=g(x), and if

$g(\bullet)$ has a unique inverse, then the probability density function of y can be computed as:

$$f_y(y) = \left| \frac{\partial y}{\partial x} \right|^{-1} \cdot f_x(x) \qquad (7)$$

The differential entropy of y is given by:

$$H(y) = -E[\ln(f_y)] = -\int_{-\infty}^{+\infty} f_y(y) \cdot \ln[f_y(y)] dy = \qquad (8)$$

$$= E\left[\ln \left| \frac{\partial y}{dx} \right| \right] - E[\ln(f_x(x))]$$

To maximize the differential entropy, we need to maximize just the first term. This maximization is carried out by a stochastic gradient ascent learning rule, where the update step can be computed as:

$$\Delta w \propto \frac{\partial H}{\partial w} = \frac{\partial}{\partial w}\left(\ln \left| \frac{\partial y}{\partial x} \right| \right) = \left(\frac{\partial y}{\partial x} \right)^{-1} \cdot \frac{\partial}{\partial w}\left(\frac{\partial y}{\partial x} \right) \qquad (9)$$

If $g(\bullet)$ becomes the logistic transfer function, of the scaled and translated input:

$$y = \frac{1}{1 + e^{-(w \cdot x + w_0)}} \qquad (10)$$

the update term can be rewritten as the update step for the weight w:

$$\Delta w \propto \frac{1}{w} + x \cdot (1 - 2y) \qquad (11)$$

and the update step for the bias weight can be computed as:

$$\Delta w_0 \propto 1 - 2y \qquad (12)$$

In the most general multivariate case, we have:

$$f_{y_1, y_2, \ldots, y_N}(y_1, y_2, \ldots, y_N) = |\underline{\underline{J}}|^{-1} \cdot f_{x_1, x_2, \ldots, x_N}(x_1, x_2, \ldots, x_N) \qquad (13)$$

where $\underline{\underline{J}}$, is the transformation Jacobian. The update step for the matrix weight becomes:

$$\underline{\underline{\Delta W}} \propto \underline{\underline{W}}^{-T} + (\underline{1} - 2\underline{y}) \cdot \underline{x}^T \qquad (14)$$

where $\underline{1}$ is a unit column vector and the update step for the bias weight vector can be computed as:

$$\underline{\underline{\Delta w}}_0 \propto \underline{1} - 2\underline{y} \qquad (15)$$

The input data are measurements of N different input sources, and, therefore, they can be referred to as a matrix $\underline{x}$, where the i-th column represents the i-th sample of the each source. The inputs of the neural network are $\underline{h} = \underline{\underline{W}} \underline{x}_s$ and $\underline{x}_s$ are called sphered data. The sphered data are computed by zero-meaning the input data $\underline{x}$ and by sphering these data with the following matrix operation:

$$\underline{x}_s = \underline{\underline{S}} \cdot \underline{x}_0 \qquad (16)$$

$$\underline{x}_0 = \underline{x} - E[\underline{x}] \qquad (17)$$

$$\underline{\underline{S}} = 2\left(\sqrt{E[\underline{x}_0 \cdot \underline{x}_0^T]} \right)^{-1} \qquad (18)$$

where $\underline{\underline{S}}$ is called sphering matrix, and it is used to speed the convergence. The infomax NN estimate the matrix $\underline{y}$, where the i-th column represents the i-th sample of the

each independent component. The architecture of the neural network is depicted in Figure 1.
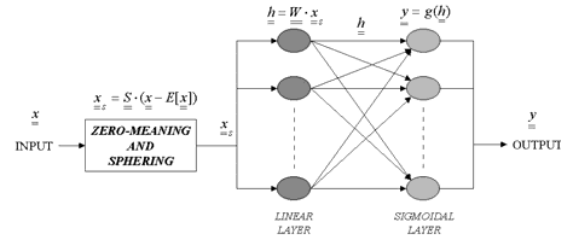
Figure 1- Architecture of the Infomax Neural Network

5.2 ICA-NN scheme based on contrast functions

The Infomax NN described in the previous Section has some limitations, both on the kind of source signals pdf and in the computational load. In this Section we will describe a different NN scheme to extract ICs that is most suitable to solve our problem. The proposed NN is also useful to cope with time-varying mixtures [Koivunen V., 2001].

The goal of ICA is to make a transform into a signal space in which the signals are statistically independent. Sometimes independence can be attained, especially in blind source separation in which the original signals are linear mixtures of independent source components and the goal of ICA is to invert the unknown mixing operation. Even when independence is not possible, the ICA transformation produces useful component signals that are non-Gaussian. The ICA allows us to approximately take into account all higher-order correlations and make the signals truly independent. Higher order statistics are needed to determine ICA expansion. In the framework of NNs, the ICA structure is that of a linear network that after learning is of the purely feed-forward type. However, during learning non-linearity must be used for separating sources. We assume here that we have a set of noisy linear mixtures representing the observed signal. By denoting with $x_k = [x_k(1), \ldots, x_k(M)]^T$ the M-dimensional kth data vector corresponding to the measurements carried out at discrete point, we can write the ICA signal model in the vector form:

$$\underline{x}_k = \underline{\underline{A}} \underline{s}_k + \underline{n}_k. \qquad (19)$$

Here $\underline{s}_k$ is the source vector consisting of the independent signal components (sources), $\underline{s}_k(i)$, i=1, N, $\underline{\underline{A}}$ = [$\underline{a}(1)$, …, $\underline{a}(N)$] is a constant MxN "mixing matrix" whose columns $\underline{a}(i)$ are the basis vectors of ICA, and $\underline{n}_k$ denotes possible corrupting noise, often omitted, because it is not possible to distinguish noise from source signals. The source separation aim is to determine $\underline{s}_k$, knowing only $\underline{x}_k$. Several assumptions must be made in ICA, in particular, only one of the source signals is allowed to have a Gaussian marginal distribution. Typically, the basis vectors $\underline{a}(i)$ are normalized to unit length and arranged according to the powers $E[\underline{s}_k(i)^2]$ in a similar way as in standard PCA. In PCA, the data model has the

same form, but the coefficient $\underline{s}_k(i)$ are required to have sequentially maximal powers (variances), and the basis vectors $\underline{a}(i)$ are constrained to be mutually orthonormal. Usually, the basis vectors of ICA are not mutually orthogonal, in order to better characterize the data. The ICA allows to determine a sparse encoding of the input vector, where histograms show a high probability of a large response as well as of no response at all. The code increases first-order redundancy (histograms) by decreasing higher-order redundancy. This redundancy transformation can be described in terms of kurtosis, that is defined by (E[.] denotes expectation):

$$k[s(i)^4] = E[s(i)^4] - 3[E[s(i)^2]]^2. \qquad (20)$$

The separation capability of various algorithms depends on the kurtosis [Ref, Kar]. It is possible to realize the estimation procedure by using a feed-forward scheme. The inputs of the NN are the M components of the vector $\underline{x}$. In the hidden layer, we have N nodes. The first layer of weights carry out a MxN whitening (and compression) of the input vector, After this, the sources are separated by means of an orthonormal matrix ($\underline{W}^T\underline{W} = \underline{I}_N$) that the NN should learn. The ICA network, firstly proposed in [Karhunen J., 1997] is shown in Figure 2. Non-linearity (i.e., hyperbolic tangent function) must be used in learning the separating matrix. The learning algorithm here used is described in [Karhunen J., 1997] and can be summarized as follows: whitening of the original data $\underline{x}$ by $\underline{v} = \underline{D}^{-1/2}\underline{E}^T\underline{x}$, where $\underline{E}$ is the matrix

of the eigenvectors of $\underline{x}$ and $\underline{D}$ is the diagonal matrix of eigenvalues that produces a starting point for an iterative process that finds vector $\underline{W}$. The learning rule is:

$$\underline{W}(k+1) = E[\underline{v}\, g(\underline{W}(k)^T\underline{v}) - g(\underline{W}(k)^T\underline{v})\,\underline{W}(k)], \quad (21)$$

where g(.) is the hyperbolic tangent. After finding W, the IC's can be found by linear combination $\underline{y} = \underline{W}^T\underline{v}$ and the mixing matrix $\underline{A}$ by $\underline{A} = \underline{E}\,\underline{D}^{1/2}\,\underline{W}$.

The use of ICA network allows us to determine the ICA separating matrix.

6. Experimental EMG data processing results

The ICA-NN scheme proposed in the previous Section has been used to extract ICs from sEMG recordings. In what follows, we will report some results that have been achieved in this study. The following Table reports the correspondence between the placements of sEMG electrodes and the related muscles. Figure 3 reports an example of the signal acquired during about 2 s of exercise (corresponding to pointing the monitor of a computer with alternatively the right and the left hand). Figure 4 reports the time-course of the 6th ICs, that appears to be mostly correlated with the 4th sEMG sensor.

$\underline{y} = \underline{W}^T\underline{V}\underline{x}$
Components of $\underline{y}$ as independent as possible

Weight matrix $\underline{Q}$ minimizes the MSE error
$E\{\|\underline{x} - \underline{Q}\,\underline{y}\|^2\}$

$\underline{W}^T$ Orthonormal
($\underline{W}^T\underline{W} = \underline{I}$) separation matrix that the network should learn
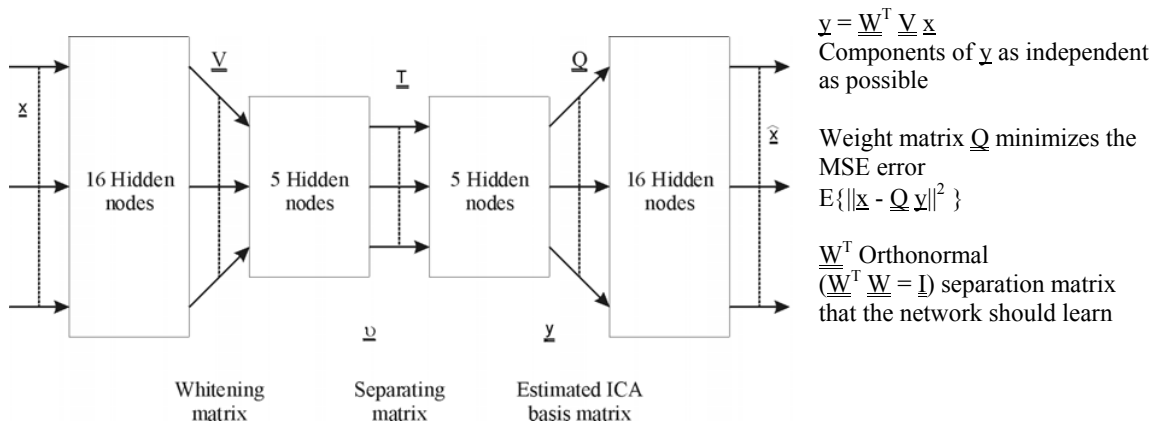
Fig. 2- The Neural Network feed-forward scheme for computing ICA.

1	SPec Superior Pectoralis	9	DBic Distal Bicep	6	MTrp Medial Trapezius	14	PWFl Proximal Wrist Flexors
2	IPec Inferior Pectoralis	10	PTri Proximal Tricep	7	LTrp Lateral Trapezius	15	DWFl Distal Wrist Flexors
3	LPec Lateral Pectoralis	11	DTri Distal Tricep	8	PBic Proximal Bicep	16	APB Abductor Pollicus Brevis
4	LDel Lateral Deltoid	12	PWEx Proximal Wrist Extensors				
5	ADel Anterior Deltoid	13	DWEx Distal Wrist Extensors				

Table 1: Correspondence between the electrode locations and the investigated muscles

Each ICs consists of a temporally independent waveform and a spatial distribution over the electrodes. The spatial distributions of the electrodes is shown on a cartoon body. The diagram has been obtained by making use of the MATLAB Toolbox for Electrophysiological Data Analysis, Version 3.2 (S. Makeig, et al, available online, http://www.cnl.salk.edu/scott/ica.html).

The electrodes are positioned according to Table 1. The colouring of each electrodes is proportional to the particular IC contributes to the electrode's raw recording. In the example, it is shown that the 6th ICs mostly contributes to the 4th electrode reading. Note the unmixing of the related components, basically activating just one electrode. Figures 6 to 8 reports the same signals for the 16th electrode and the 16th ICs. In this case, the 16th component mainly activates the same electrode.

Measuring the ICs of sEMG will provide a more reliable and robust measure of motor performance than interpreting the activity of each individual muscles in isolation [Jung T.P., 2001].

There are practical advantages of separating the sEMG signals into temporally ICs, namely, the ICs are less susceptible to changes in position of the electrodes, and therefore more suitable for serially monitoring performance; the ICs are, in addition, more likely to correspond to brain activations [Jung T.P., 2001], by looking for common cortical influences in the muscle activity.

As previously mentioned, the experiment described in the present Section have been carried out by using a Neural Network scheme to implement ICA. It is, of course, possible, to use different techniques to implement ICA, however, it could be demonstrated that the use of a NN approach is equivalent to other approaches, like maximum likelihood estimation. The NN scheme is most suitable to achieve hardware implementation.
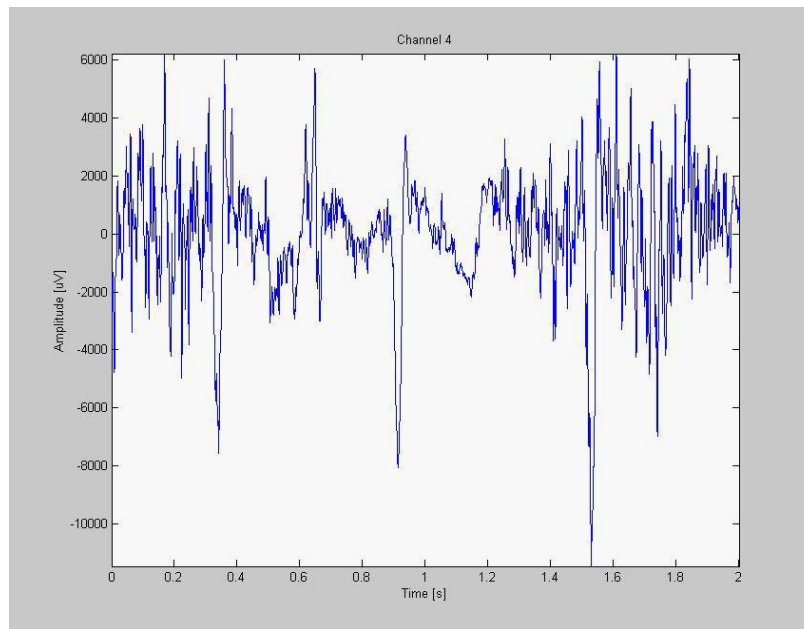
Figure 3: Raw EMG recording from the 4th electrode
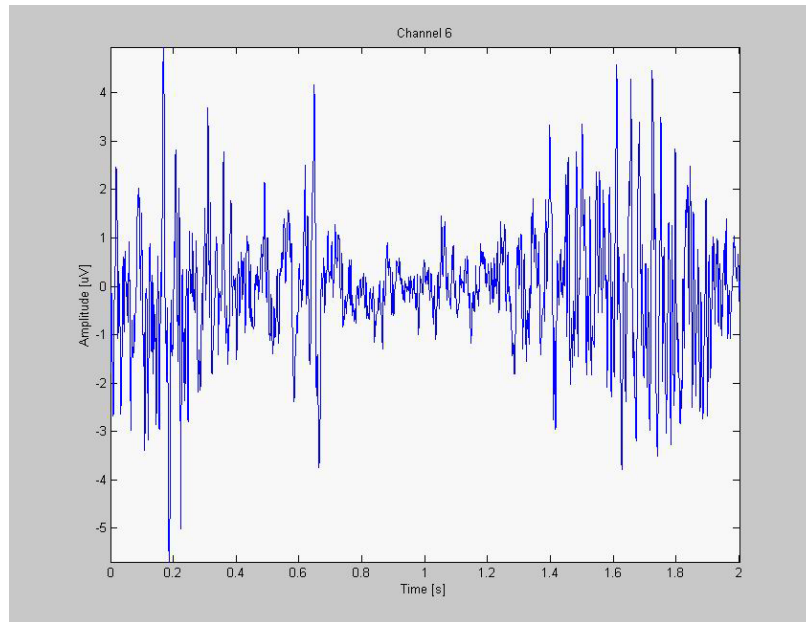
Figure 4: Time-course of the 6th extracted ICs
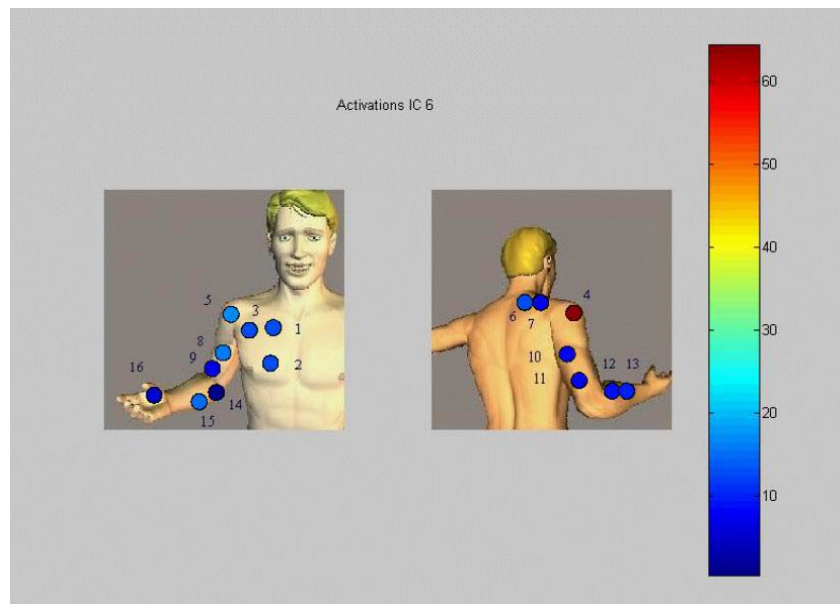
Figure 5: Spatial distribution of the activations corresponding to the 6th ICs
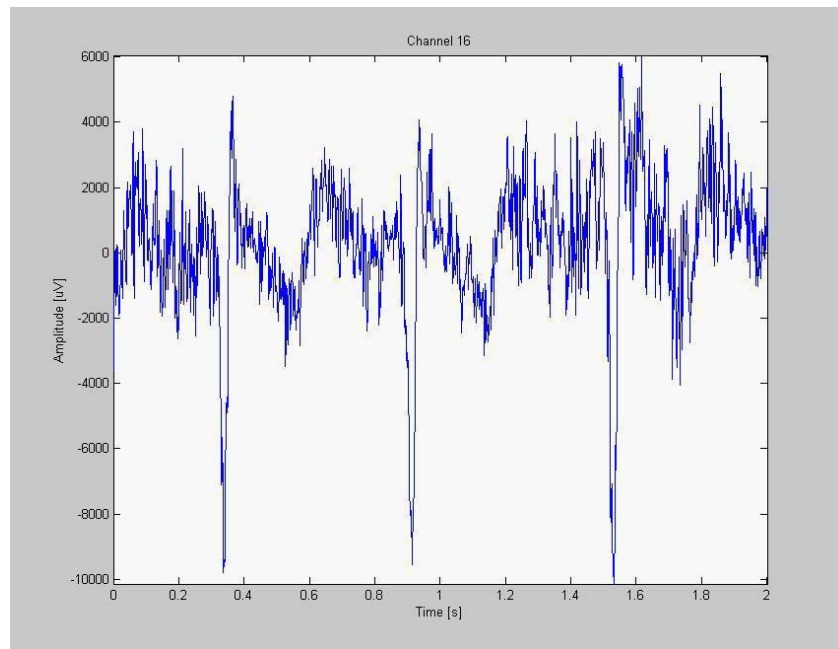
Figure 6: Raw EMG recording from the 16th electrode
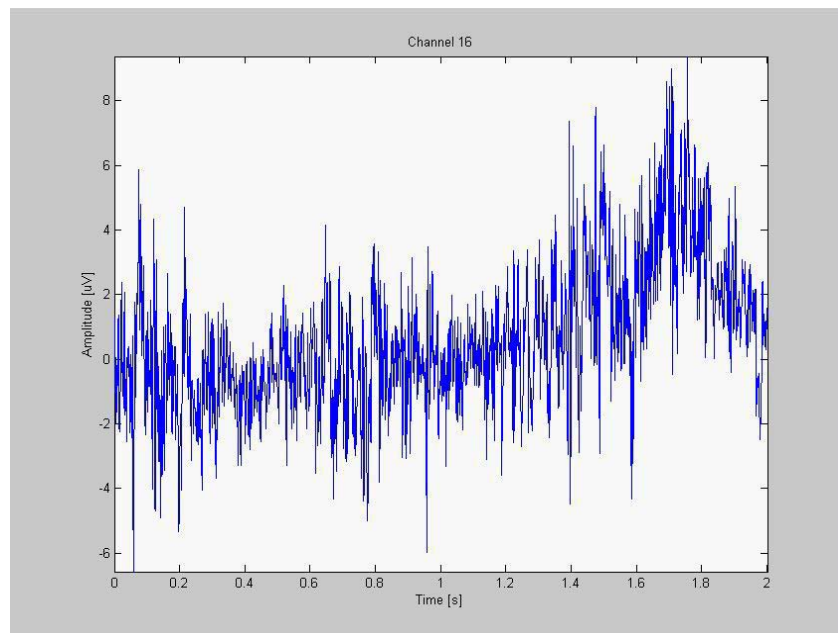
Figure 7: Time-course of the 16th extracted ICs
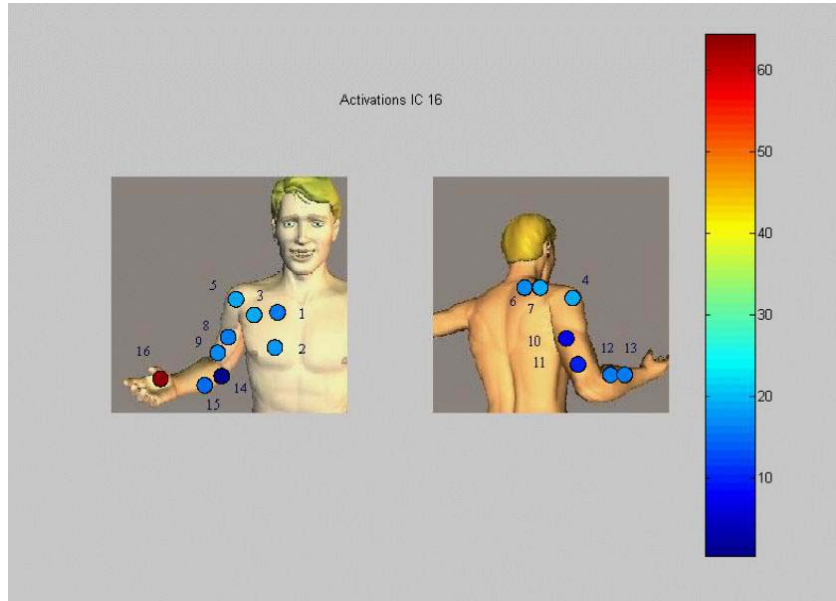
Figure 8: Spatial distribution of the activations corresponding to the 16th ICs

7. Treatment of non-stationarity

The extraction of ICs is based on the assumption of stationarity among different trials of the same experiment. In the practice, for such sEMG data, this is a hardly acceptable assumption. We would like now to propose a time-frequency approach to the analysis of sEMG data (or their ICs counterparts) that allows to cope with signal non-stationarity. The sEMG is indeed non-stationary as its statistical properties change over time. The MUAPs (Motor Unit Action Potentials) are transients that exist for a short period of time: for that reason, time-frequency methods are useful to characterize the localized frequency content of each MUAP. The use of a time-frequency representation also allows, in principle, to detect the onset of sub-movements, according to what we explained in the previous Sections. We have carried out the wavelet analysis in both the time domain of sEMG and of the ICs, in order to show that this kind of analysis should be carried out on the original space (the IC space is generated by already making a stationarity assumption).

The wavelet transform also guarantees to possibility of not specifying in advance the key signal features and the optimal basis functions needed to project the signal in order to highlight the features. An orthogonal wavelet transform is characterized by two functions:

1) the scaling function,

$$\phi(x) = \sqrt{2} \sum_{k \in Z} h(k) \phi(2x - k) \qquad (22)$$

and 2) its associated wavelet:

$$\psi(x) = \sqrt{2} \sum_{k \in Z} g(k) \phi(2x - k) \qquad (23)$$

where g(k) is a suitable weighting sequence (function).

The sequence h (k) is the so-called refinement filter. The wavelet basis functions are constructed by dyadic dilation (index j) and translation (index k) of the mother wavelet:

$$\psi_{jk} = 2^{-j/2} \psi(x/2^{-j} - k) \qquad (24)$$

The sequences h and g can be selected such that $\left\{\psi_{jk}\right\}_{(jk) \in Z^2}$ constitutes an orthonormal basis of L2, the space of finite energy functions. This orthogonality permits the wavelet coefficients $d_j(k) = \langle f, \psi_{jk} \rangle$ and the approximation coefficients $c_j(k) = \langle f, \phi_{jk} \rangle$ of any function f(x) to be obtained by inner product with the corresponding basis functions. In practice, the decomposition is only carried out over a finite number of scales J. The wavelet transform with a depth J is then given by:

$$f(x) = \sum_{j=1}^{J} \sum_{k \in Z} d_j(k) \psi_{jk} + \sum_{k \in Z} c_J(k) \phi_{Jk} . \qquad (25)$$

In the present study, we shall use the WT in order to derive a set of features that can reveal singularity of the signal (corresponding to the onset of activity of single muscles) and to detect the precursors of the non-stationarity. A set of features derived from the inspection of the scale-dilation plane have been used as input vector of an auto-associative NN that is able to alarm the user about modification of the energy content of the spectrum. The features are extracted by considering the correspondence between singularities of a function and local maxima of its wavelet transform. A singularity corresponds to pairs of modulus maxima across several scales. Feature extraction is accomplished by the computation of the singularity degree (peakiness), i.e.,

the local Lipschitz regularity, which is estimated from the wavelet coefficients decay [Mallat S., 1992, Arkidis N.S., 2002].

Figures 9 and 10 reports the amplitude sEMG signal for channel 4th, and the wavelet transform obtained by using Daubechies 1 and 4 mother wavelet. The modulus maxima plots have been drawn and a thresholding operator is used in order to reduce the number of effective wavelet coefficients needed to represent the original functions. Once the features have been extracted by inspecting the modulus maxima plot, we can use the corresponding nonzero coefficients in order to predict the raising of non stationarity. A MLP NN with an input layer of corresponding size acts as a bottleneck network (the output size is the same of the input one, while the hidden layer size is considerably reduced). The NN fed by the wavelet coefficients computes the estimation of the corresponding wavelet coefficients at the output: a reconstruction error is computed. If the error overcomes a prescribed threshold level, the non-stationarity signal is activated and the following trials are used to compute a novel matrix (ICs) weights. The use of a MLP-NN is not obliged to ensure accuracy or success in the reconstruction; for example, a different compression scheme could be used, like the Singular Value Decomposition. The bottleneck layer is, in principle, able to work as principal component extractor, but the idea here is to build a compressed representation which is deliberately redundant. The reconstruction error could be sub-optimal with respect to different schemes, but optimality comes at the expenses of quite low fault tolerance. Finally, the MLP NN can be implemented easily in a FPGA hardware chip. A typical case of non-stationarity is the onset of fatigue. The Figure 11 describes how the activation intervals [Micera S., 2001] of the muscles during the exercise cycle are determined starting from the ICs.

The standard approach to determine on-off activation patterns is to process each epoch by means of a double threshold statistical detector [Bonato P., 1998, Balestra G., 2001] to obtain the muscle detection intervals. We have compared the results achieved by our method with the one described and we have found an improvement of about 20% in the performance.
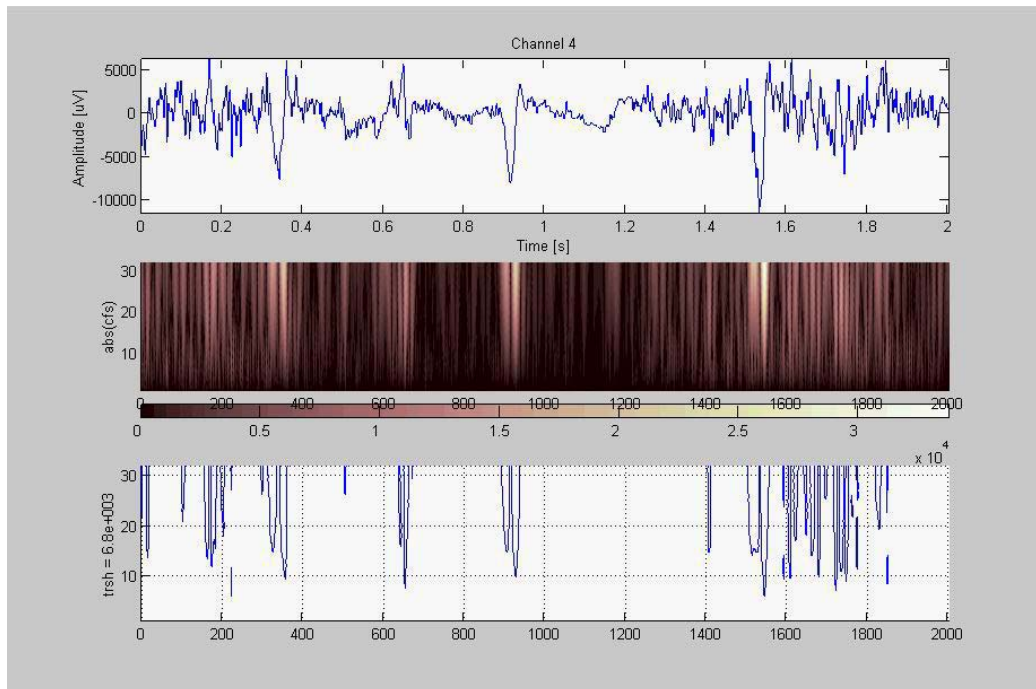
Figure 9: The wavelet transform of the 4th sEMG channel (mother wavelet, Daubechies 1): the raw data recording (top), the plot of the absolute values of the WT coefficients (middle) and the modulus maxima extracted (bottom). A thresholding is applied to suppress WM that are not of interest. White colour corresponds to high value of the coefficients. If one uses a wavelet with one vanishing moment, then the bottom plot corresponds to the maxima of the smoothed first-order derivative of the function.
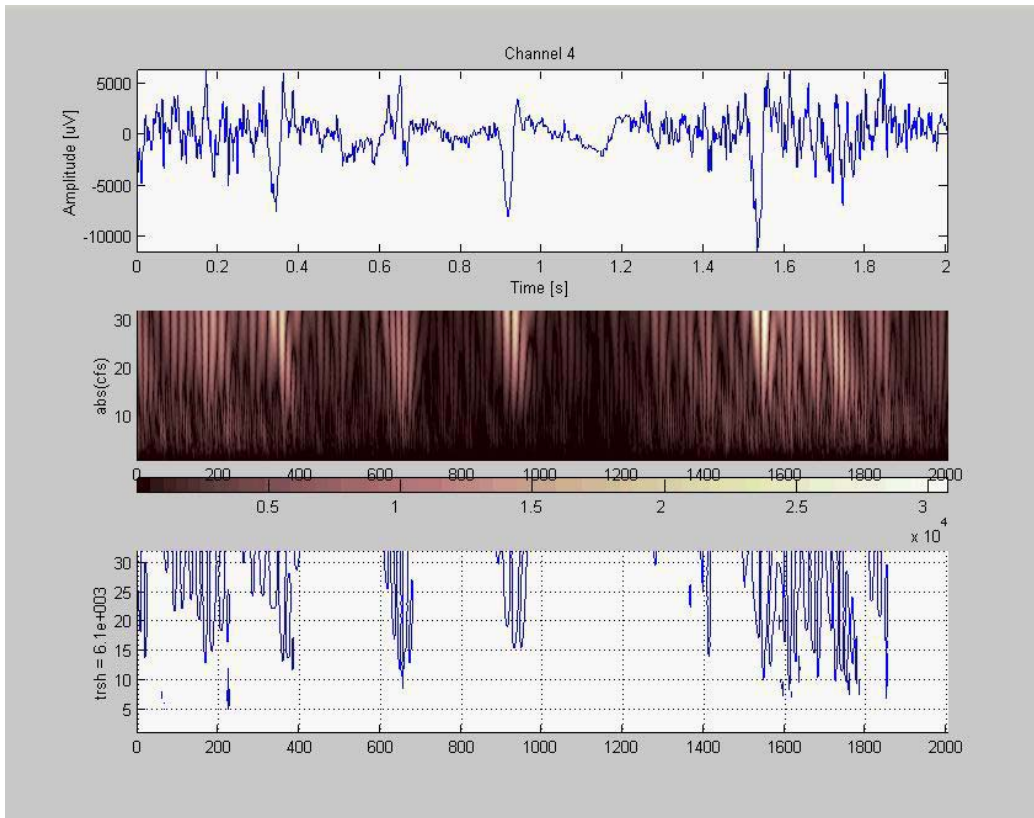
Figure 10: The wavelet transform of the 4th sEMG channel (mother wavelet, Daubechies 4): the raw data recording (top), the plot of the absolute values of the WT coefficients (middle) and the modulus maxima extracted (bottom). White colour corresponds to high value of the coefficients. A wavelet function with 4 vanishing moments is used.
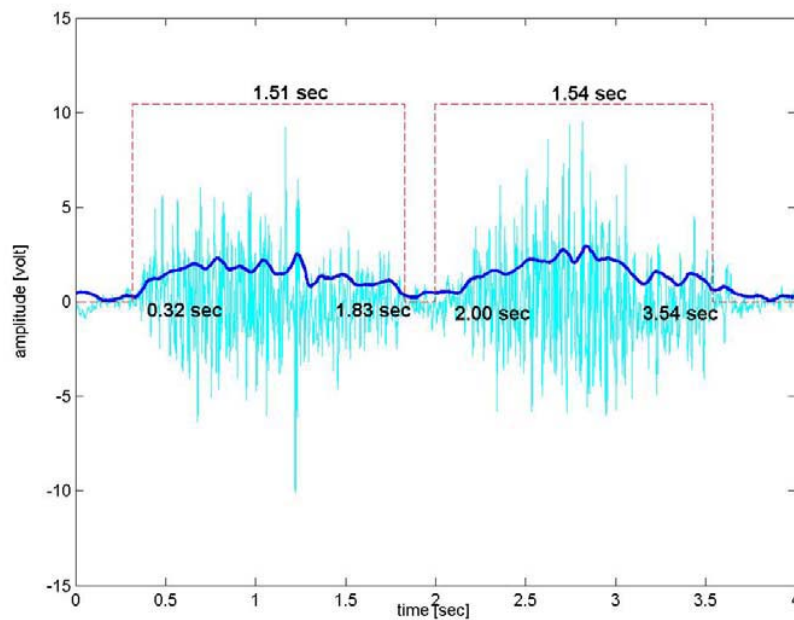
Figure 11: The determination of the activation intervals (the wavelet envelope is used).

8. Conclusion

The paper proposed the use of some NNs to process experimental electrical data derived from non-invasive sEMG experiments. The original (raw) data have been analysed by a neural IC processor aiming to obtain signals that can be easily correlated to cortical activity. The assumption of stationarity is then relaxed in order to cope with time-varying mixing systems, more adherent to the biophysical problem at hand. An auto-associative NN exploits the features obtained by wavelet transforming the raw data for making a quick and efficient prediction of non-stationarity. The results we have shown can be considered just as preliminary to solve the difficult problem.

Acknowledgment

REFERENCES

Akay M., *Time-frequency and Wavelets in Biomedical Signal Processing*, Piscataway, NJ, IEEE Press, 1997.

Balestra G., Frassinelli S., Knaflitz M. Molinari F., *Time-Frequency Analysis of Surface Myoelectric Signals Durino Athletic Movement*, IEEE Engineering in Medicine and Biology, Special Issue on Monitoring Muscles in Motion, Vol. 20, No.6, Nov/Dec 2001.

Bonato P., D'Alessio T., and Knaflitz M., *A statistical method for the measurement of muscle activation intervals from surface myoelectric signal during gait*, IEEE Trans. Biomed. Eng., Vol.45, pp.287-299,1998.

Arkidis N.S., Abel E.W., Faster A., *Interscale Wavelet Maximum: A Fine to Coarse Algorithm for Wavelet Analysis of the EMG Interference Pattern*, IEEE Trans. On Biom. Eng., Vol. 49, No. 4, pp.337-344, April, 2002.

Bell A.J. and Sejnowski T.J., *An Information-Maximization Approach to Blind Separation and Blind Deconvolution,* Neural Computation*, Vol.*7, pp.1129-1159, 1995.

Bell A.J. and Sejnowski T.J., *Learning the higher-order structure of a natural sound*, Network Computation in Neural Systems, Vol.7, 1996.

Harris C.M., *On the optimal control of behaviour: A stochastic perspective*, J. Neurosci. Meth, Vol. 83, pp. 73-88, 1998.

IEEE Engineering in Medicine and Biology, Special Issue on Monitoring Muscles in Motion, Vol. 20, No.6, Nov/Dec 2001.

Jung T.P., Makeig S., Lee T.W., McKeown M.J., Brown G., Bell A.J. and Sejnowski T.J, *Independent Component Analysis of Biomedical Signals* – The 2nd Int'I Workshop on Independent Component Analysis and Signal Separation, 2000.

Jung T.P., Makeig S., McKeown M.J., Bell A.J., Lee T.W., Sejnowski T.J., *Imaging brain dynamics using independent component analysis*. P IEEE. 89(7): 1107-22, 2001

Karhunen J., Oja E., et al, *A class of neural networks for independent component analysis*, IEEE Trans. on Neural Networks, Vol. 8, N. 3, pp.486-504, 1997.

Koivunen V., Enescu M., Oja E., *Adaptive algorithm for blind separation from noisy time-varying mixtures*, Neural Computation, Vol.13, pp.2339-2357, 2001.

Lee T.W., *Independent Component Analysis: Theory and Applications*, Kluwer Academic Publishers, Boston, 1998.

Lee T.W., Girolami M., and Sejnowski T.J., *Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources*, Neural Computation, Vol. 11, N.2, pp. 417-441, 1999.

Linsker R., *An Application of the Principle of Maximum Information Preserving to Linear Systems*, In Advances in Neural Information Processing Systems 1.

Mallat S. and Hwang W.L., *Singularity detection and processing with wavelets*, IEEE Trans. Inform. Theory, Vol.38, pp. 617-643, 1992.

McKeown M.J., *Phasic and Tonic Coupling between EEG & EMG Revealed with ICA*, J. Clin. Neurophysiology, 2000a, in press

McKeown M.J., *Cortical activation related to arm movement combinations*, Muscle Nerve, Vol 9, Suppl. 9.4, 2000b.

McKeown M.J., Torpey D.C., Gehm W.C.. *Non-Invasive Monitoring of Functionally Distinct Muscle Activations during Swallowing Clinical Neurophysiology* 2002 (in press).

Micera S., Vannozzi G., Sabatini A.M., Dario P., *Improving Detection of Muscle Activation intervals*, IEEE Engineering in Medicine and Biology, Vol. 20, No. 6 Nov-Dec 2001 pp. 38-46.

Sejnowski T.J., *Making smooth moves*, Nature, Vol.394, Aug 1998.

Francesco Carlo Morabito was born and educated in Italy, where he worked in the University since 1989 after a three-year experience as a researcher in the electronic industry. Since December 2001, he serves as Dean of the Faculty of Engineering of the "Mediterranea" University of Reggio Calabria, Italy, where he teaches (full professor) Electrical Engineering and Neural Networks. He was a member of INNS since 1992 and served as Board of Governors member for the last three years (2000-2002). He served as Secretary of INNS in 2002 and he is now serving as Secretary. He chairs the Regional Italy SIG since 1997, taught and co-ordinated Tutorials at the IJCNN1999, 2000, 2001. Since 2002, he served as an action editor for Neural Networks. He also serves as both steering and scientific committee of Italian Society for Neural Networks (SIREN). He chaired the Eduardo Caianiello Scientific Prize. He is a senior Member of IEEE. He taught at the EPFL, Lausanne, Switzerland at the Ph.D. courses in 1999 and worked as a referee for both international research projects and Ph.D. thesis in the field of Neural Networks. He was general chair of AMSE-ISIS'97 and '98, and AEDEM 2001, member of the scientific committees of many international conferences in Europe, Asia, USA and Africa. He was a visiting researcher at the Max-Planck Institut fuer Plasmaphysik, NET Team, Asdex-Upgrade, ITER Joint Team, Garching bei Muenchen, Germany for some years. He was scientific responsible for various national and international scientific research project.

Listed in Who's Who in the World, Who's Who in Science and Engineering; International Biographical Centre. He is the author of 1 book, 3 book chapters, and more than 150 scientific papers. He is the recipient of the 2002 "Henry Coanda" Prize from Rumanian Academy for his scientific activity in the field of Neural Networks.

Maurizio Campolo was born in Reggio Calabria, Italy. He has been working since 1993 as computer science researcher at the University Mediterranea of Reggio Calabria. His research interests are in the field of Neural Networks and Information Technology.

He is the author of more than 20 scientific papers and worked as a technician for some national research projects. He was member of the scientific committe of international conference SSCC98 in Africa and member of the organizing committe of AMSE-SIS97, ENDE98 and HBD2000 in Italy. Since 1993 he is the network administrator and webmaster of the Faculty of Engineering. He also worked with NeuroLab.

A Combined State Space Formulation/Equivalent Circuit and Neural Network Technique for Modeling of Embedded Passives in Multilayer Printed Circuits

X. Ding, J.J. Xu, M.C.E. Yagoub [+] and Q.J. Zhang

Department of Electronics, Carleton University, Ottawa, Ontario, Canada K1S 5B6
[+] School of Information Technology and Engineering, University of Ottawa, Ottawa, Ontario, Canada, K1N 6N5

Abstract : In this paper, we present a new approach for modeling the high-frequency effects of embedded passives in multilayer printed circuits, utilizing state space equations or equivalent circuit together with neural network techniques. In this approach, the neural network based model structure is trained using full wave electromagnetic (EM) data. The resulting embedded passive models are accurate and fast, can be used in both frequency/time domain simulators. Examples of embedded resistor and capacitor models demonstrate that the combined model can accurately represent EM behavior in microwave/RF circuit design. In high-level circuit design, we applied our combined EM based neural models for signal integrity analysis and design of multilayer circuit to illustrate that the geometrical parameters can be continuously adjusted by using neural network techniques. Optimization and Monte-Carlo analysis are performed showing that the combined models can be efficiently used in place of computationally intensive EM models of embedded passives to speed up circuit design.

I. INTRODUCTION

The drive in the electronics industry for manufacturibility-driven design and time-to-market demands powerful and efficient computer-aided design (CAD) techniques. As the signal frequency increase, the dimensions of embedded passives in multilayer circuits become a significant fraction of signal wavelength. The conventional time/frequency domain electrical models for the components are not accurate anymore. As EM effects play an important role in microwave/RF circuit design, models with continuous physical/geometrical information must include EM effects [1]. Furthermore, the need of optimization and statistical analysis taking into account process variations and manufacturing tolerances in the components makes it extremely important that the component models should be accurate and fast so that the design solutions can be achieved feasibly and reliably.

Recently, artificial neural network (ANN) modeling approach has been studied for microwave modeling and design [2-4]. The neural models can be as fast as empirical models and as accurate as detailed physics models.

For high-level circuit design, the component models should be continuously varied both with frequency, geometrical and/or electrical parameters. Therefore, modeling techniques that can provide such continuous variations are essential and ANN models exactly meet for these requirements. They are continuous, multi-dimensional and can easily handle nonlinearities in problem behaviours. Neural network techniques have been widely used to model variety of microwave device/circuits such as transmission line components [5], bends [6], vias [7], spiral inductors [8], and FET devices [5, 9].

Embedded passives represent an emerging technology area that has the potential for increased reliability, improved electrical performance, size shrinkage, and reduced cost. The conventional approach for circuit and system design requires equivalent circuits to capture the response of embedded passives [10]. But the equivalent circuit method may not be accurate enough to reflect high frequency EM effects. Recently, neural network techniques have been introduced to model frequency behavior of embedded passives [1]. However, such ANN models, trained to learn S-parameters data, cannot be used directly into time-domain circuit simulation and optimization. Our target was to develop passive ANN based models from EM data that can be used directly in both time and frequency domain circuit design.

In this paper, we present a novel approach to model high-frequency effects of embedded passives in multilayer printed circuits based on combined equivalent circuit or state space theory together with neural networks. Our combined model is a hierarchical structure with two levels. In the lower level, a neural network maps the geometrical/physical parameters of the passive component into coefficient matrices of state equations or lumped component values of a given equivalent circuit. In the higher level, we export the coefficient matrices into the state space equation or component values into the equivalent circuit to compute the EM response in either frequency or time domain circuit design. The accurate and fast ANN based embedded passive models are trained from full wave EM data. Our method combines existing modeling techniques and recent neural network approaches to efficiently perform simulation and optimization. Based on neural network techniques, geometrical/physical parameters become design variables to improve circuit performance and reduce design/manufacture cost.

In Section II, the problem for neural modeling of embedded passives is summarized. In Section III, we present the combined equivalent circuit and neural network (EC-NN) modeling approach. The combined State space equation and neural network (SSE-NN) modeling approach is presented in section IV. The method is demonstrated by embedded resistor and capacitor examples in section V. Signal

integrity of multilayer circuit, which includes SSE-NN models of embedded passives, is used to demonstrate the application of the model for circuit simulation. Optimization and Monte-Carlo analysis are performed showing that the geometry inputs can be continuously adjustable by using our combined models and the model evaluation is much faster than computationally intensive physical/EM model of passives in microwave design.

II. Embedded Passives Neural Modeling: Problem Statement

Let x represent a N_x-vector containing parameters of a microwave device/circuit, e.g., length and width of an embedded resistor, or thickness and dielectric constant of an embedded capacitor. Let $\hat{y}$ represent a $N_{\hat{y}}$-vector containing the responses of the component under consideration, e.g., Y- or S-parameters. The physics/EM relationship between $\hat{y}$ and x can be highly nonlinear and multi-dimensional. The theoretical model for this relationship may not be available, or theory may be too complicated to implement, or the theoretical model may be computationally too intensive for online microwave design and repetitive optimization (e.g., 3D full-wave EM analysis inside a Monte-Carlo statistical design loop). We aim to develop a fast and accurate neural model by teaching/training a neural network to learn the embedded passive problem. Let the neural network model be defined as

$$\hat{y} = f_{ANN}(x, w) \tag{1}$$

where w represents the parameters inside the neural network also called as the weight vector. The most widely used neural network structure is the feedforward multilayer perceptrons (MLP) [2, 5, 7] where neurons are grouped into layers, and each neuron in a layer acts as a smooth switch that produces a response between low and high state according the weighted responses of all neurons from the preceding layer. The neural network structure allows the ability to represent multidimensional nonlinear input/output mappings accurately, and to evaluate $\hat{y}$ from x quickly. To enable a neural network to represent a specific microwave x - $\hat{y}$ relationship, we first train the neural network to learn the microwave data pairs (x_i, d_i) where x_i is a sample of x, d_i is a vector representing the $\hat{y}$ data generated from microwave simulation or measurement under given sample x_i, and i is the sample index. For training purpose, we define an error function E(w) as

$$E(w) = \frac{1}{2} \sum_{i \in Tr} \sum_{k=1}^{N_{\hat{y}}} \left(f_{ANN\,k}(x_i, w) - d_{ki} \right)^2 \tag{2}$$

where d_{ki} is the kth element of d_i, $f_{ANN\,k}(x_i, w)$ is the kth output of the neural network for input sample x_i and Tr is an index set of all training samples. The objective of neural network training is to adjust neural network connection

weights w such that E(w) is minimized. A trained neural model can then be used online during microwave design stage providing fast model evaluation replacing original slow model from EM simulators. The benefit of the neural model is especially significant when the model is repetitively used in design processed such as optimization, Monte-Carlo analysis, and yield optimization [11]. However, MLP models, trained to learn S-parameters data, cannot be used directly into time-domain circuit simulation and optimization. We aim to develop a fast and accurate combined model, which uses equivalent circuit and neural network, through EM data to learn the embedded passive problem.

Let $g_p = \{R, L, C\}$ be a N_p-vector containing the values of lumped components of a given equivalent circuit topology T_p. We use a neural network to represent g_p as

$$g_p = f_{ANN}(x, w) \tag{3}$$

and then the combined model can be defined as

$$\hat{y}(\omega) = f_f \left(T_p \left(f_{ANN}(x, w) \right), \omega \right) \tag{4}$$

$$y(t) = f_t \left(T_p \left(f_{ANN}(x, w) \right), t \right) \tag{5}$$

where ω is the angular frequency, $\hat{y}(\omega)$ and $y(t)$ are the combined model response in frequency and time domain respectively, e.g., $\hat{y}(\omega)$ can be S- or Y-parameters and $y(t)$ can be the currents $i(t)$ and voltages $v(t)$ of a two port embedded passive. Therefore, a combined model realizes the x - $\hat{y}/y$ relationship through a MLP and then equivalent circuit.

III.   Combined Equivalent Circuit and Neural Network (EC-NN) Modeling Approach

A. Introduction of EC-NN Model

A number of fast equivalent circuit models of embedded passive components are available. In [12, 13], two methods are presented for developing equivalent circuit using optimization methods. Synthesize lumped element equivalent circuit from rational function is presented in [10]. Although we can get equivalent circuit in many ways from measured or simulated EM data, an equivalent circuit only represents a fixed embedded passive structure. If the embedded passive's geometrical/physical parameters need to be changed, we have to re-generate a new equivalent circuit to match it.

In this paper, EC-NN model exploits neural network features to accurately predict element values of equivalent circuit based on geometrical/physical parameters. EC-NN model, motivated by [14], is a hierarchical model with two levels. At the lower level, a neural model maps the geometrical/physical parameters of the passive component into lumped component values of a given equivalent circuit. At the higher level, we supply these values into the

equivalent circuit to compute the EM response in frequency or time domain circuit design.

B. EC-NN Model development

We utilize an existing equivalent circuit and combine it with a MLP together to make the model automatically as function of geometrical/physical parameters. The EM data of embedded passives, which consists of geometrical/physical parameters as inputs and real/imaginary parts of S-parameters as outputs, are generated by simulation or measurement.

To create data for neural network training, we extract the lumped component values based on the existing equivalent circuit through a set of measured/simulated sample pairs of EM data. Considering some measurement noise in the EM data, the parameter extraction criterion for each set of input geometry is defined as an optimization objective function as

$$\underset{\boldsymbol{g}_p}{Min} \sum_{i \in T_r} \sum_{k=1}^{N_g} \left\| f_f\big(T_p\big(\boldsymbol{g}_p\big), \omega\big) - d_{ki} \right\| \qquad (6)$$

This objective function shows that adjusting the lumped component values $\boldsymbol{g}_p$ to map the S-parameters of high-frequency response of the equivalent circuit best match the EM data in the interested frequency bandwidth. Due to the complexity of the error function, iterative algorithms are used to explore the lumped component values. The optimization algorithms we used are gradient and quasi-Newton methods. We collected the lumped component values versus geometrical/physical parameters as neural network training data. We teach/train a MLP to learn the relationships between equivalent circuit component values and geometry. Let $\boldsymbol{g}_{pi}$ be a vector representing $\boldsymbol{g}_p$ data under given sample $\boldsymbol{x}_i$. The error function is defined as

$$\mathrm{E}(\boldsymbol{w}) = \sum_{i \in Tr} \sum_{k=1}^{N_p} \left\| f_{ANN\,k}(\boldsymbol{x}_i, \boldsymbol{w}) - g_{p\,ki} \right\| \qquad (7)$$

where g_{pki} is the kth element of $\boldsymbol{g}_{pi}$. After training, the MLP can accurately calculate the component values varied with continous geometry for the given equivalent circuit. The last step is to export the EC-NN model into a user defined simulation format, e.g., SPICE sub-circuit netlist format. The EC-NN model includes two sections. The first section is a set of mathematical equations to represent the internal structure of neural network that calculate the lumped component values based on different geometry/physical inputs. The second section is the updated equivalent circuit, which receives the element values from MLP outputs. In a circuit simulator, the EC-NN model will be feed by geometrical/physical parameters as inputs. The MLP automatically calculates the element values in a user defined equivalent circuit and supply the values into the equivalent circuit to represent EM behavior in frequency and time domain.

IV.   Combined State Space Equation and Neural Network (SSE-NN) Modeling Approach

A. Formulation in Frequency-Domain

Topology of equivalent circuit is a sensitive factor of the combined model accuracy and a given topology may not be suitable for different geometry and frequency range. In order to develop an accurate model, which can be represented more efficiently in both time and frequency domain simulation, we proposed the combined SSE-NN modeling approach.

EM data of an embedded passive can be collected depending on different geometrical/physical parameters from full wave EM simulation/measurement. For a given frequency range, we can use transfer functions (polynomial rational functions) to represent the electrical behavior (e.g., admittance $\mathbf{Y}$ matrix) of the embedded passives. For any two-port embedded passives, the following three transfer functions are adequate to represent Y_{11}, Y_{21}, and Y_{22}, respectively.

$$H_1(s) = \frac{b_0 + b_1 s + \cdots + b_{n-1} s^{n-1} + b_n s^n}{a_0 + a_1 s + \cdots + a_{n-1} s^{n-1} + s^n} \qquad (8.a)$$

$$H_2(s) = \frac{d_0 + d_1 s + \cdots + d_{n-1} s^{n-1} + d_n s^n}{a_0 + a_1 s + \cdots + a_{n-1} s^{n-1} + s^n} \qquad (8.b)$$

$$H_3(s) = \frac{c_0 + c_1 s + \cdots + c_{n-1} s^{n-1} + c_n s^n}{a_0 + a_1 s + \cdots + a_{n-1} s^{n-1} + s^n} \qquad (8.c)$$

where $s = j\omega$, and n is the number of effective order of the passive. Let us define a real coefficient vector, as $\boldsymbol{g}_v = \{a_0, a_1, \ldots a_{n-1}; b_0, b_1, \ldots b_n; c_0, c_1, \ldots c_n; d_0, d_1, \ldots d_n\}$. Using space-mapping concept [6], a relationship exists between the coefficients and geometrical/physical parameters. However, the relationship would be highly nonlinear and too complicated. Therefore, we utilize neural network features to learn the highly nonlinear relationship between the coefficients and geometrical/physical parameters.

In the coefficient parameter extraction procedure, we used gradient and quasi-Newton optimization algorithms to enforce $H(s)$ to best match EM data. The objective function was defined as

$$\underset{\boldsymbol{g}_v}{Min} \sum_{i \in Tr} \sum_{k=1}^{3} \left\| H_k(\boldsymbol{g}_v, \omega) - d_{ki} \right\| \qquad (9)$$

and we use a neural network to learn the relationship between coefficient vector $\boldsymbol{g}_v$ and EM input parameters $\boldsymbol{x}$,

$$\boldsymbol{g}_v = f_{ANN}(\boldsymbol{x}, \boldsymbol{w}). \qquad (10)$$

We used the center point of input space as the initial point to optimize the coefficient vector values.

B. State Space Equation for Time-Domain Simulation

Using coefficients g_v in (8), we can define

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 & \vdots & \vdots & \ddots & \vdots & \vdots \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix}_{2n \times 2n}$$

$$B = \begin{bmatrix} 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 \end{bmatrix}^T_{2n \times 2} \quad D = \begin{bmatrix} b_n & d_n \\ d_n & c_n \end{bmatrix}_{2 \times 2} \quad (11)$$

$$C = \begin{bmatrix} b_0 - a_0 b_n & \cdots & b_{n-1} - a_{n-1} b_n & d_0 - a_0 d_n & \cdots & d_{n-1} - a_{n-1} d_n \\ d_0 - a_0 d_n & \cdots & d_{n-1} - a_{n-1} d_n & c_0 - a_0 c_n & \cdots & c_{n-1} - a_{n-1} c_n \end{bmatrix}_{2 \times 2n}$$

to form the state space equation,

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \quad (12)$$

where $x(t)$ is a vector of internal states, u and y are vectors of the input and output signals, e.g., input voltages and output currents of the embedded passive respectively. Our combined model can be then implemented into a time domain circuit simulator using the state space equation (12) or into a frequency domain circuit simulator using (8).

C. Stability and passivity

To assure stability requirement in time domain simulation, the poles of the combined SSE-NN model need to be on left half plane (LHP) [15]. To enforce all the poles of the transfer functions of embedded passives to be into LHP, we added a set of constraints in the parameter extraction as

$$P_{even-order} = \prod_{i=1}^{T} P_{2i} \; ; \text{ where } P_{2i} = (s^2 + k_{2i}s + k_{3i}) \text{ and } T = n/2, \text{ if}$$

$k_{2i} > 0 \;\&\; k_{3i} > 0$; all of real and complex roots in LHP.

$$P_{odd-order} = P_1 \cdot \prod_{i=1}^{T} P_{2i} \; ; \text{ where } P_1 = (s + k_1) \text{ and } T = (n-1)/2, \text{ if}$$

$k_1 > 0 \;,\; k_{2i} > 0 \;\&\; k_{3i} > 0$; all of real and complex roots in LHP.

where $k = \{k_1, k_{21}, k_{31}, \ldots k_{2T}, k_{3T}\}$ is a vector of components that lead to elements in the matrix A. For example, in a 3rd order combined model, the denominator coefficients are defined as $a_0 = k_1 \cdot k_3$; $a_1 = k_1 \cdot k_2 + k_3$; and $a_2 = k_1 + k_2$, respectively.

The criterion for passivity can be defined if the eigenvalues of $G = \text{Re}\{Y\}$ are positive [15, 16]. This condition can be assured if $y_{12} y_{21} \leq y_{11} y_{22}$, where the y_{jk} ($j,k = 1,2$) are real parts of the Y matrix elements. It has been used as an optimization constraint in the g_v parameter extraction procedure.

The above criterions are added in the parameter extraction to ensure that the rational functions not only accurately represent EM behavior but also enforce the time domain model to be stable and passive.

D. Structure of the Combined SSE-NN Model

Our combined SSE-NN model is a hierarchical structure with two levels. At the lower level, a neural network maps the geometrical/physical parameters into g_v vectors. At the higher level, we insert the coefficient vectors into the state equations to compute the EM response in frequency or time domain simulation. Fig. 1 shows the structure of the combined model for both EC-NN and SSE-NN.

For circuit CAD tools in time domain, we export our SSE-NN into SPICE sub-circuit format. The lower neural network will be described by a set of mathematical equations, which calculate the coefficient values based on different geometrical/physical parameters and pass them into higher level. The equivalent circuit can be generated from (11) and (12).
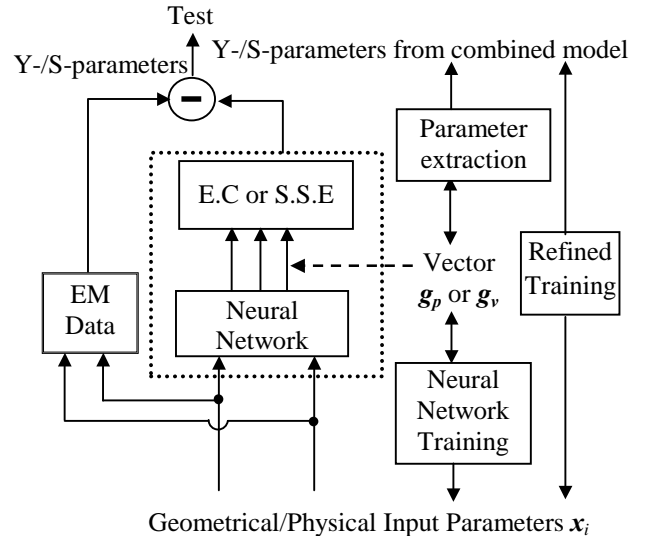
Figure 1. Structure of the combined EC-NN and SSE-NN models illustrating the model development process and the testing phase. E.C. and S.S.E. represent equivalent circuit and state space equation respectively.

E. Combined SSE-NN Model Development

EM data has component's geometrical/physical parameters and frequency as inputs and S-parameters as outputs. The next phase is parameter extraction, which is carried out for each geometry over the entire frequency range. The objective here is to determine the coefficient values that best fit the original EM data. Different geometrical parameter values and their corresponding coefficient values are then re-arranged into neural network training data. A 3-

layer MLP neural network is trained using quasi-Newton algorithm in *NeuroModeler* [17]. For any given geometrical dimensions of the component within the range of the training data, the trained MLP can predict the elements of vector g_v. We combine the state equation with the neural model using our hierarchical setup to obtain the overall combined model. The inputs to the combined model are the geometrical dimensions of the embedded component. The intermediate outputs of the model are the corresponding coefficient vector values. The final outputs of the combined model are component's EM behavior, e.g., S-parameters. In the test phase, an independent set of test data containing S-parameters versus new geometrical parameter values (i.e., never seen during training) is generated using the EM simulator. This data is used to test the accuracy of the combined model. In the final phase, we formulate the combined model into a set of mathematical expressions to be directly used to carry out high-level circuit design in time-domain simulators.

V. Examples

In order to demonstrate the proposed modeling approach, we developed embedded resistors and capacitors in EC-NN and SSE-NN models. We applied the SSE-NN models in signal integrity of multilayer circuit design to efficiently perform optimization and statistic analysis.

A. Embedded Resistor

Accurate modeling of EM behaviors of embedded passive used in high-speed multilayer printed circuit board is important for efficient high-speed circuit design. In this example, a combined EC-NN model of an embedded resistor shown in Fig. 2 is developed. The EM data of the embedded resistor is automatically generated from EM simulation of Sonnet [18]. Length (L) and width (W) are used as inputs. The outputs are real and imaginary parts of S_{11} and S_{21} in the EM data. Fig. 3 shows the structure of the EC-NN model for the embedded resistor, which includes an equivalent circuit and a 3-layer MLP neural network.
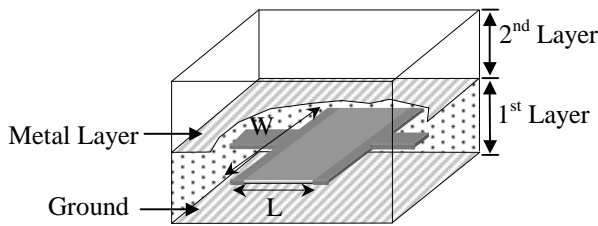
Figure 2. 3-D physical structure of embedded resistor.

The neural network is trained to learn the relationship about the input geometry and the four lumped component values (R1, R2, C1, C2). After the MLP is well trained, it can accurately calculate the component values based on any within geometrical/physical parameters for the given equivalent circuit even the parameters was never used in training. Testing is performed by comparing the outputs of the overall EC-NN model and EM data, shown in Fig. 4(a).

Because the neural network can provide the accurate component values continuously varied with geometry for the equivalent circuit, the combined EC-NN model can be in place of the computationally intensive physical/EM model to efficiently provide EM effects in optimization and statistic design.
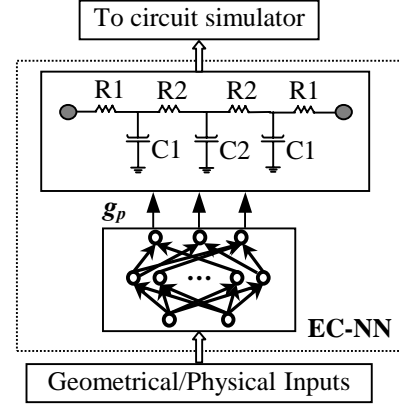
Figure 3. The structure of the combined EC-NN model for embedded resistors. The equivalent circuit is user defined.
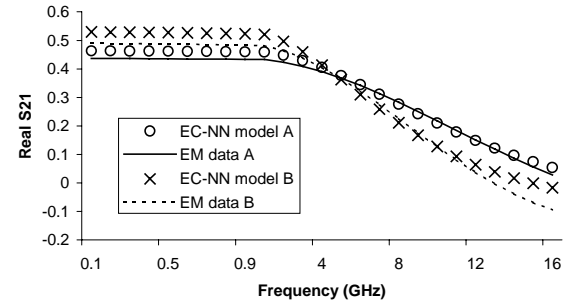
(a)
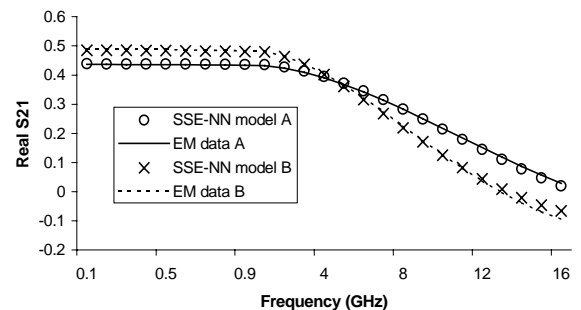
(b)

Figure 4. Comparison of real part of S_{21} of embedded resistor EC-NN model outputs (a) or SSE-NN model outputs (b) and independent EM data which was never used in training. Curves A are generated based on W = 1.346 and L = 0.279 mm. Curves B are generated based on W = 0.99 and L = 0.254 mm.

The test error of combined EC-NN model is 5.8%. Further improvement of accuracy requires new topology of equivalent circuit. Instead of using human based trial and error process, we use the proposed SSE-NN modeling method. As the equivalent circuit for the embedded resistor uses three capacitors, a 3rd order transfer function can express the behavior of the embedded resistor in the SSE-NN model.

Table I shows the model test error, which we achieved, based on various orders of state equations in SSE-NN modeling development. The test error demonstrated that the optimal number of internal states is three. In 4th order model, the additional internal state could not play an important role in the EM behavior representation. However, more coefficients are needed in transfer function, more freedom in parameter extraction and neural network training.

The best results are obtained with the 3rd order SSE-NN model. The agreement between 3rd order SSE-NN model and EM data is achieved even though the independent testing data was never seen in training, shown in Fig. 4(b). To verify stability and passivity, the three LHP poles of the embedded resistor model are -1.4411 and -0.0144 $\pm$ j0.0539, and the passivity condition is satisfied as shown in Fig. 5.

Table I. Comparison of resistor SSE-NN model with different order formulations.

Order	Test Error
2nd	1.59%
3rd	1.12%
4th	2.38%
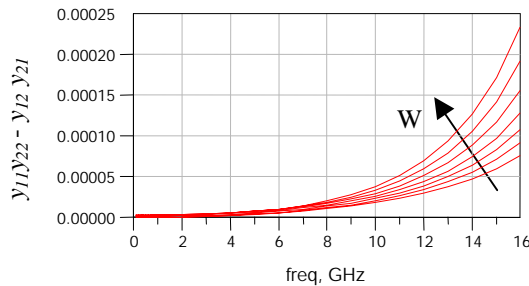
Figure 5. The 3rd order SSE-NN model in frequency-domain simulation and y_{jk} $(j,k = 1,2)$ are real part of the **Y** matrix elements. The W is swept from 0.952mm to 1.397mm.

B. Embedded Square Capacitor

The physical structure of an embedded square capacitor is shown in Fig. 6. The input parameters include length (L), capacitor dielectric constant (ε_{rcap}), and frequency. Real and imaginary parts of S-parameters are generated from 3D full wave EM simulator, Ansoft-HFSS [19]. Fig. 7 shows the equivalent circuit used in our combined EC-NN model for the embedded capacitor.
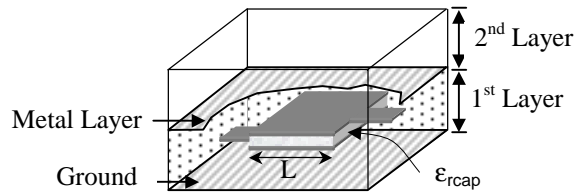
Figure 6. 3-D physical layout of embedded capacitor.

The neural network is trained to learn the embedded capacitor inputs and lumped component values. For example, L1=0.035nH, C1= 1.135pF, C2=0.537pF when L=0.736mm and ε_{rcap}=17.5. The S-parameter comparison between the EC-NN model and original EM data is shown in Fig. 8(a). Table II illustrates the different test error, which we achieved, based on varied order formulas in SSE-NN modeling development.

The optimal transfer function is 3rd order to represent the EM based capacitor. Testing is performed by comparing the outputs of combined SSE-NN models and EM data. The agreement between our 3rd order SSE-NN model and EM data is obtained even though the independent testing data was never seen in training, shown in Fig. 8(b).
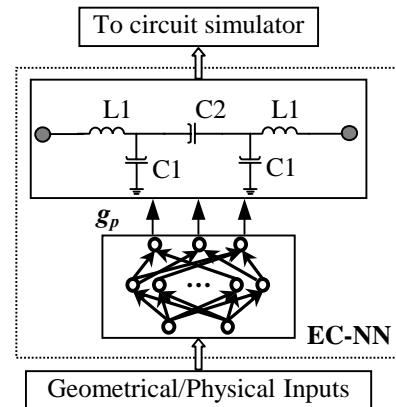
Figure 7. The combined EC-NN model structure for embedded capacitor. The equivalent circuit is user defined.

(a)

(b)
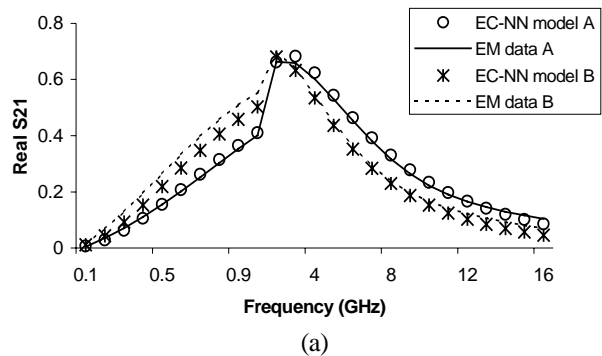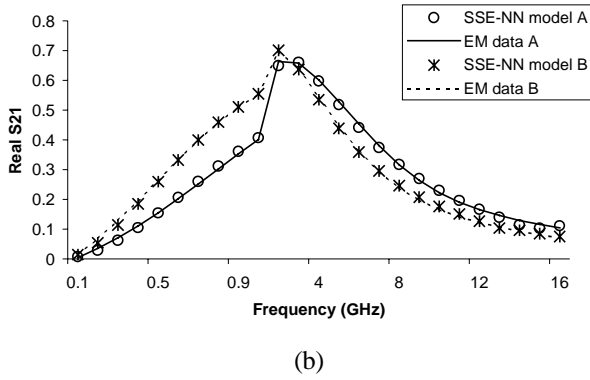
Figure 8. Comparison of real part of S_{21} of embedded capacitor EC-NN model outputs (a) or SSE-NN model outputs (b) and independent EM data. Curves A and B are generated based on inputs L = 0.736mm and L = 0.787mm respectively.

Table II. Comparison of capacitor SSE-NN model with different order formulations.

Order	Test Error
2nd	2.20%
3rd	1.67%
4th	2.57%

3. Signal Integrity Example

To further confirm the validity of the proposed combined model in time-domain, we plugged the above resistor and square capacitor SSE-NN models into a time-domain simulator, i.e., *Hspice* [20] to perform circuit simulation and optimization including geometrical and physical parameters of the embedded passives. The models help to achieve a convenient link between EM behaviors and high-level circuit design, improving design accuracy and efficiency. In this paper, we use signal integrity of multilayer circuit as shown in Fig. 9, where the length and width of embedded resistor and length and dielectric constant of embedded capacitor are adjustable.
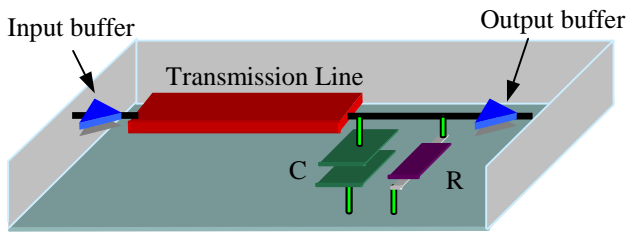
Figure 9. Three dimensional illustration of signal integrity of multilayer circuit with embedded resistor and capacitor.

In optimization process, whenever optimization changes the geometry, the corresponding combined models are called with the new geometrical dimensions as inputs. From output comparison, as shown in Fig. 10, the output curves have been improved in terms of distortion and time delay.
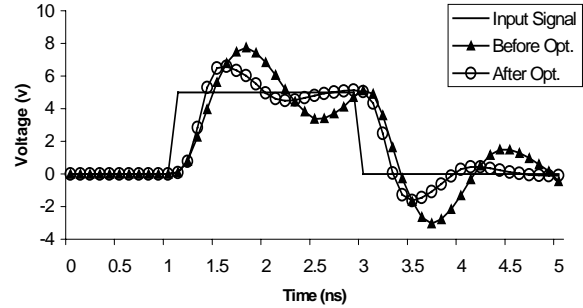
Figure 10. Comparison of signal from input buffer, and output signals before and after combined SSE-NN models optimization.

The optimization used 136 iterations including repetitive evaluation of combined SSE-NN models to reach the criteria of the optimization goal and the total computation time based on our combined SSE-NN models is 3.75minutes. The results show that the combined models provide possibility to adjust the geometry of embedded passives in high-frequency circuit design. Because we used neural models to learn the nonlinear relationship between geometry and coefficient vectors, the geometry becomes variable in circuit design.

We also performed statistical analysis of the signal integrity circuit with our SSE-NN models in a three-coupled transmission line circuit as shown in Fig. 11. Monte-Carlo analysis of signal integrity curves with geometrical parameters as statistical design variables are shown in Fig. 12. The total simulation time for 500 output curves based on the geometry tolerance around the nominal design center is 8.24 minutes using proposed neural models by *Hspice*. However, the required time of Ansoft-HFSS for 500 different geometry is more than 8 hours. The proposed combined models retain the advantages of neural network learning, speed, and accuracy, and provide EM effects in high-level circuit design.
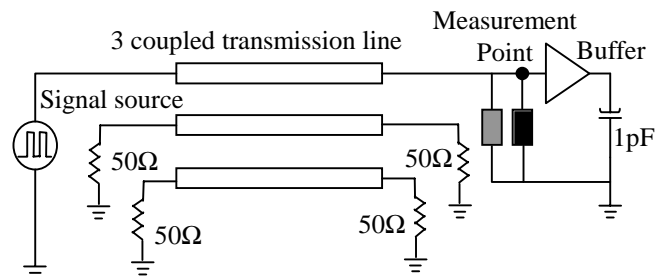
Figure 11. The three coupled transmision line circuit.
　　■ : EM capacitor SSE-NN model;
　　■ : EM resistor SSE-NN model
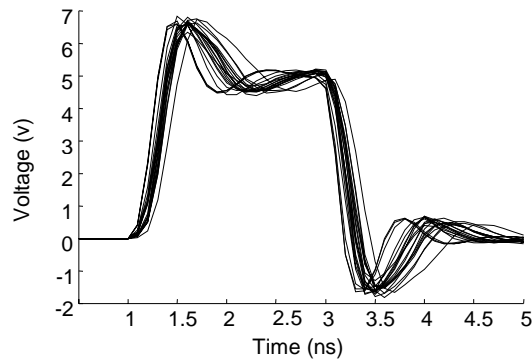
Figure 12. Output of Monte-Carlo analysis of the 3-coupled transmission line example using SSE-NN models of embedded passives. Here 20 randomly chosen curves are shown out of 500 simulations of the circuit of Fig. 11.

VI. Conclusions

In this paper, we presented a new method for modeling embedded passives suitable for both frequency and time domain simulation. The combined models, which utilize neural network and equivalent circuit or state space equation techniques, are developed from EM data.

The accuracy of the combined EC-NN model will depend on the equivalent circuit in the combined model for the entire frequency range. If the accurate and reliable equivalent circuit is available, EC-NN will be generated efficiently, because the number of lumped elements in equivalent circuit is less than the number of coefficient values in state space equations.

In combined SSE-NN model development, we automatically generate an accurate solution for modeling embedded passives, avoiding human based trial and error process in conventional approach. The combined SSE-NN modeling technique acts as a bridge to combine slow physical EM model and fast equivalent circuit model to together. In high-speed circuit design, the combined neural models allow geometrical/physical parameters to become design variables in circuit simulation. Therefore, manufacture geometrical tolerance can be taken into account in circuit design efficiently and accurately.

VII. Acknowledgments

VIII. References

[1]   Q.J. Zhang, M.C.E. Yagoub, X. Ding, D. Goulette, R. Sheffield, and H. Feyzbakhsh, "Fast and accurate modeling of embedded passives in multi-layer printed circuits using neural network approach", *Elect. Components & Tech. Conf.*, San Diego, CA, May 2002, pp. 700 – 703.

[2]   Q.J. Zhang, and K.C. Gupta, *Neural Networks for RF and Microwave Design*, Artech House, Norwood, MA, 2000.

[3]   A.H. Zaabab, Q.J. Zhang, and M.S. Nakhla, "A Neural network modeling approach to circuit optimization and statistical design", *IEEE Trans. Microwave Theory Tech.*, vol. 43, pp. 1349-1358, 1995.

[4]   P. Burrascano, S. Fiori, and M. Mongiardo, "A review of artificial neural networks applications in microwave computer-aided design", *Int. J. RF and Microwave CAE*, vol. 9, pp. 158-174, 1999.

[5]   F. Wang, V.K. Devabhaktuni, and Q.J. Zhang, "A hierarchical neural network approach to the development of a library of neural models for microwave design," *IEEE Trans. Microwave Theory Tech.*, vol. 46, pp. 2391-2403, 1998.

[6]   J. Bandler, M. Ismail, J. Rayas-Sanchez, and Q. Zhang, "New directions in model development for RF/microwave components utilizing artificial neural networks and space mapping," *IEEE APS Int. Symp. digest*, Orlando, FL, July 1999, pp. 2572-2575.

[7]   P.M. Watson and K.C. Gupta, "EM-ANN models for microstrip vias and interconnects in dataset circuits," *IEEE Trans. Microwave Theory Tech.*, vol. 44, pp. 2495-2503, 1996.

[8]   G.L. Creech, B.J. Paul, C.D. Lesniak, T.J. Jenkins, and M.C. Calcatera, "Artificial neural networks for fast and accurate EM-CAD of microwave circuits," *IEEE Trans. Microwave Theory Tech.*, vol. 45, pp. 794-802, 1997.

[9]   Y. Harkouss, J. Rousset, H. Chehade, E. Ngoya, D. Barataud, and J.P. Teyssier, "Modeling microwave devices and circuits for telecommunications system design," *IEEE Int. Conf. Neural Networks*, Anchorage, Alaska, May 1998, pp. 128-133.

[10] K.L. Choi and M. Swaminathan, "Development of model libraries for embedded passives using network synthesis," *IEEE Trans. Circuits and Systems*, vol. 47, pp. 249-260, 2000.

[11] X. Ding, B. Chattaraj, M.C.E. Yagoub, V.K. Devabhaktuni, Q.J. Zhang, "EM based statistical design of microwave circuits using neural models", *Int. Symp. on Microwave and Optical Technology*, Montreal, Canada, June, 2001, pp. 421-426.

[12] J. Zhao, R.C. Frye, W.W.-M. Dai, and K.L. Tai, "S parameter-based experimental modeling of high Q MCM inductor with exponential gradient learning algorithm," *IEEE Trans. Comp., Packag., Manufact. Technol. B,* vol. 20, pp. 202-210, 1997.

[13] R. Poddar and M.A. Brooke, "Accurate high speed empirically based predicative modeling of deeply embedded gridded parallel plate capacitors fabricated in a multilayer LTCC process," *IEEE Trans. Comp., Packag., Manufact. Technol.,* vol. 22, pp. 26-31, 1999.

[14] K. Shirakawa, M. Shimizu, N. Okubo, and Y. Daido, "A large signal characterization of an HEMT using a multilayered neural network," *IEEE Trans. Microwave Theory Tech.*, vol. 45, pp. 1630-1633, 1997.

[15] S.H. Min and M. Swaminathan, "Efficient construction of two-port passive macromodels for resonant networks," *IEEE Int. Conf. Elec. Packag.,* Cambridge, MA, Oct 2001, pp. 229-232.

[16] B. Gustavsen and A. Semlyen, "Enforcing passivity for admittance matrices approximated by rational functions," *IEEE Trans. Power system,* vol. 16, pp. 97-104, 2001.

[17] *NeuroModeler v.* 1.3, Q.J. Zhang, Dept. of Electronics, Carleton University, Ottawa, Canada.

[18] *Sonnet v* 7.0, Sonnet Software, Liverpool, NY, USA.

[19] *Ansoft HFSS v.*8.0, Ansoft Corp., Pittsburg, PA, USA.

[20] *Hspice v.*2001.2, Avant! Corp., Fremont, CA, USA.

Xiaolei Ding received the B.Eng. degree in Electrical Engineering from North China Institute of Technology, Taiyuan, China, in 1993, and the M.A.Sc degree in electronics from Carleton University, Ottawa, Canada, in 2002. He is currently working towards the Ph.D. degree in the Department of Electronics, Carleton University, Ottawa, Canada. His research interests include neural network modeling of EM effects for embedded passives and their applications in CAD for high-speed/high-frequency circuits.

Jianjun Xu was born in Aug. 1975, in Liaoning, China. He received the B.Eng. degree in Electrical and Electronics Engineering from Tianjin University, Tianjin, China in 1998. He is currently a Ph.D student in the Department of Electronics, Carleton University, Ottawa, ON, Canada. His research interests include neural networks, modeling and their applications in computer-aided design for electronics circuits.

Mustapha C.E. Yagoub received the *Diplôme d'Ingénieur* degree in Electronics and the *Magister* degree in Telecommunications, both from the *Ecole Nationale Polytechnique*, Algiers, Algeria, in 1979 and 1987 respectively. In 1994, he obtained his Ph.D. degree from the *Institut National Polytechnique*, Toulouse, France. He was with the Institute of Electronics of the *Université des Sciences et de la Technologie Houari Boumédienne*, Algiers, Algeria, first as an assistant during 1983-1991 and then as an assistant professor during 1994-1999. From 1999 to 2001, he was at the Department of Electronics, Carleton University, Ottawa.

He is currently an assistant professor in the School of Information Technology and Engineering, University of Ottawa, Ottawa, Canada. His research interests include neural networks, CAD of linear and nonlinear microwave devices and circuits, and applied electromagnetics. He has over 80 publications in international journals and conferences. He is the first author of "*Conception de circuits linéaires et non linéaires micro-ondes*" (Cépadues Ed., Toulouse, France, 2000). Dr. Yagoub is a member of the Association of Professional Engineers of the Province of Ontario, Canada.

Qi-Jun Zhang received the B.Eng. Degree from the East China Engineering Institute, Nanjing, China, in 1982, and the Ph.D. Degree in Electrical Engineering from the McMaster University, Hamilton, Canada, in 1987.

He was with the Systems Engineering Institute, Tianjin University, Tianjin, China, during 1982-1983. From 1988 to 1990, he was with the Optimization Systems Associates Inc. (OSA), Dundas, Ontario, Canada, developing advanced microwave optimization software. He joined the Department of Electronics, Carleton University, Ottawa, Canada in 1990, where he is presently a Professor. His research interests are neural network and optimization methods for high-speed/high-frequency circuit design. He has authored and co-authored over 150 papers in the area. He is a co-author of *Neural Networks for RF and Microwave Design* (Artech House, Boston, 2000), a co-editor of *Modeling and Simulation of High-Speed VLSI Interconnects* (Kluwer, Boston, 1994), a contributor to *Analog Methods for Computer-Aided Analysis and Diagnosis* (Marcel Dekker, New York, 1988), a guest co-editor for a Special Issue on High-Speed VLSI Interconnects for *the International Journal of Analog Integrated Circuits and Signal Processing* (Kluwer, Boston, 1994), and twice a guest editor for the Special Issues on Applications of ANN to RF and Microwave Design for *the International Journal of RF and Microwave CAE* (Wiley, New York, 1999 and 2002). Dr. Zhang is a member of the Association of Professional Engineers of the Province of Ontario, Canada.

One-vs-One Multiclass Least Squares Support Vector Machines for Direction of Arrival Estimation

Judd A. Rohwer[1] and Chaouki T. Abdallah[2]

[1]Sandia National Laboratories, P.O. Box 5800 MS-0986, Albuquerque, NM, 87185-0986 Email: jarohwe@sandia.gov

[2]Department of Electrical and Computer Engineering, MSC01 1100

University of New Mexico, Albuquerque, NM, 87131-001 Email: chaouki@eece.unm.edu

Abstract— **This paper presents a multiclass, multilabel implementation of Least Squares Support Vector Machines (LS-SVM) for DOA estimation in a CDMA system. For any estimation or classification system the algorithm's capabilities and performance must be evaluated. This paper includes a vast ensemble of data supporting the machine learning based DOA estimation algorithm. Accurate performance characterization of the algorithm is required to justify the results and prove that multiclass machine learning methods can be successfully applied to wireless communication problems. The learning algorithm presented in this paper includes steps for generating statistics on the multiclass evaluation path. The error statistics provide a confidence level of the classification accuracy.**

I. INTRODUCTION

Machine learning research has largely been devoted to binary and multiclass problems relating to data mining, text categorization, and pattern recognition. Recently, machine learning techniques have been applied to various problems relating to cellular communications, notably spread spectrum receiver design, channel equalization, and adaptive beamforming with direction of arrival estimation (DOA). In our research we present a machine learning based approach for DOA estimation in a CDMA communication system [1]. The DOA estimates are used in adaptive beamforming for interference suppression, a critical component in cellular systems. Interference suppression reduces the multiple access interference (MAI) which lowers the required transmit power. The interference suppression capability directly influences the cellular system capacity, i.e., the number of active mobile subscribers per cell.

Beamforming, tracking, and DOA estimation are current research topics with various technical approaches. Least mean square estimation, Kalman filtering, and neural networks [2],[3],[4], have been successfully applied to these

problems. Many approaches have been developed for calculating the DOA; three techniques based on signal subspace decomposition are ESPRIT, MUSIC, and Root-MUSIC [1].

Neural networks have been successfully applied to the problem of DOA estimation and adaptive beamforming in [4], [5], [6]. New machine learning techniques, such as support vector machines (SVM) and boosting [7], perform exceptionally well in multiclass problems and new optimization techniques are published regularly. These new machine learning techniques have the potential to exceed the performance of the neural network algorithms relating to communication applications.

The machine learning methods presented in this paper include subspace based estimation applied to the sample covariance matrix of the received signal. The one-vs-one multiclass LS-SVM algorithm uses both training data and received data to generate the DOA estimates. The end result is an efficient approach for estimating the DOAs in CDMA cellular architecture [1].

This paper is organized as follows. Section II presents the system models for an adaptive antenna array CDMA systems. A review of binary and multiclass machine learning methods is presented in Section III, along with background information on the LS-SVM algorithm. Section IV includes a brief review of classic DOA estimation algorithms and the elements of a machine learning based DOA estimation algorithm. Section V presents a one-vs-one multiclass LS-SVM algorithm for DOA estimation and simulation results are presented in Section VI. Section VII includes a comparison between standard DOA estimation algorithms and our machine learning based algorithm.

II. SYSTEM MODELS

This section includes an overview of system models for the received signal and adaptive antenna arrays designs. All notation is described below and is consistently used throughout the paper.

A. Received Signal at Antenna Array output

The baseband signal, $\mathbf{r}_A(t)$, from the antenna array is

$$\mathbf{r}_A(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}_r(t). \tag{1}$$
$$\mathbf{A} = \begin{bmatrix} \mathbf{a}(\theta_1) & \mathbf{a}(\theta_2) & \ldots & \mathbf{a}(\theta_L) \end{bmatrix} \tag{2}$$
$$\mathbf{a}(\theta_l) = \begin{bmatrix} 1 & e^{-jk_l} & e^{-j2k_l} & \ldots & e^{-j(D-1)k_l} \end{bmatrix}^T \tag{3}$$
$$\mathbf{s}(t) = \begin{bmatrix} s_1(n) & s_2(n) & \ldots & s_L(n) \end{bmatrix}^T \tag{4}$$
$$s_l(t) = \sqrt{p_{t_i}(t)}q^l b_i(t), \text{ for path } l, \tag{5}$$

where $\mathbf{r}_A(t)$ is the received signal of mobile i, $\mathbf{A}$ is a $D \times L$ array steering vector for D antenna elements and L transmission paths, $\mathbf{s}(t)$ is the $L \times 1$ received baseband signal at the output of the matched filter, $\mathbf{a}(\theta_l) = \begin{bmatrix} 1 & e^{-jk_l} & \ldots & e^{-j(D-1)k_l} \end{bmatrix}^T$ is the $D \times 1$ steering vector, $k_l = \frac{\nu w_c}{c}\sin\theta_l$, v is the spacing between antenna elements, w_c is the carrier frequency, c is the velocity of propagation, θ_l is the direction of arrival of the l signal, $p_{t_i}(t)$ is the transmit signal power from mobile i, q_i^l is the attenuation due to shadowing from path l, $b_i(t)$ is the data stream of mobile i, and $\mathbf{n}_r(t)$ is the additive noise vector.

To ease the complexity of the notation the terms relative to the multiple paths are combined as

$$\mathbf{z}_i = \sum_{l=1}^{L} \mathbf{a}(\theta_l)q_i^l. \tag{6}$$

In [8] $\mathbf{z}_i$ is defined as the spatial signature of the antenna array to the ith mobile.

III. Support Vector Machines - Background

A major machine learning application, pattern classification, observes input data and applies classification rules to generate a binary or multiclass labels. In the binary case, a classification function is estimated using input/output training pairs,$(\mathbf{x}_i, y_i)$ $i = 1 \ldots n$, with unknown probability distribution, $P(\mathbf{x}, y)$,

$$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathbb{R}^N \times Y, \tag{7}$$
$$y_i = \{-1, +1\}. \tag{8}$$

The estimated classification function maps the input to a binary output, $f : \mathbb{R}^N \to \{-1, +1\}$. The system is first trained with the given input/output data pairs then the test data, taken from the same probability distribution $P(\mathbf{x}, y)$, is applied to the classification function. For the multiclass case $Y \in \mathbb{R}^G$ where Y is a finite set of real numbers and G is the size of the multiclass label set. In multiclass classification the objective is to estimate the function which maps the input data to a finite set of output labels $f : \mathbb{R}^N \to S(\mathbb{R}^N) \in \mathbb{R}^G$

Support Vector Machines (SVMs) were originally designed for the binary classification problem. Much like all machine learning algorithms SVMs find a classification function that separates data classes, with the largest margin,

using a hyperplane . The data points near the optimal hyperplane are the "support vectors". SVMs are a nonparametric machine learning algorithm with the capability of controlling the capacity through the support vectors.

A. Kernel Functions

The kernel based SVM maps the input space into a higher dimensional feature, F, space via a nonlinear mapping

$$\Gamma : \mathbb{R}^N \to F \tag{9}$$
$$\mathbf{x} \mapsto \Gamma(\mathbf{x}). \tag{10}$$

The data does not have the same dimensionality as the feature space since the mapping process is to a non-unique generalized surface [9]. The dimension of the feature space is not as important as the complexity of the classification functions. For example, in the input space, separating the input/output pairs may require a nonlinear separating function, but in a higher dimension feature space the input/output pairs may be separated with a linear hyperplane. The nonlinear mapping function $\boldsymbol{\Gamma}(\mathbf{x}_i)$ is related to kernel, $\Bbbk(\mathbf{x}, \mathbf{x}_i)$ by

$$\Gamma(\mathbf{x}) \cdot \boldsymbol{\Gamma}(\mathbf{x}_i) = \Bbbk(\mathbf{x}, \mathbf{x}_i). \tag{11}$$

Four popular kernel functions are the linear kernel, polynomial kernel, radial basis function (RBF), and multilayer perceptrons (MLP).

$$\text{linear, } \Bbbk(\mathbf{x}, \mathbf{x}_i) = \mathbf{x} \cdot \mathbf{x}_i \tag{12}$$
$$\text{polynomial, } \Bbbk(\mathbf{x}, \mathbf{x}_i) = ((\mathbf{x} \cdot \mathbf{x}_i) + \boldsymbol{\theta})^{\mathbf{d}} \tag{13}$$
$$\text{RBF, } \Bbbk(\mathbf{x}, \mathbf{x}_i) = \exp\left(\frac{-\|\mathbf{x} - \mathbf{x}_i\|^2}{\sigma^2}\right) \tag{14}$$
$$\text{MLP, } \Bbbk(\mathbf{x}, \mathbf{x}_i) = \tanh(\kappa(\mathbf{x} \cdot \mathbf{x}_i) + \boldsymbol{\theta}) \tag{15}$$

The performance of each kernel function varies with the characteristics of the input data. Refer to [10] for more information on feature spaces and kernel methods.

B. Binary Classification

In binary classification systems the machine learning algorithm generate the output labels with a hyperplane separation where $y_i \in [-1, 1]$ represents the classification "label" of the input vector $\mathbf{x}$. The input sequence and a set of training labels are represented as $\{\mathbf{x}_i, y_i\}_{i=1}^{n}$, $y_i = \{-1, +1\}$. If the two classes are linearly separable in the input space then the hyperplane is defined as $\mathbf{w}^T\mathbf{x} + b = 0$, $\mathbf{w}$ is a weight vector perpendicular to the separating hyperplane, b is a bias that shifts the hyperplane parallel to itself. If the input space is projected into a higher dimensional feature space then the hyperplane becomes $\mathbf{w}^T\boldsymbol{\Gamma}(\mathbf{x}) + b = 0$.

The SVM algorithm is based on the hyperplane definition [11],

$$y_i\left[\mathbf{w}^T\boldsymbol{\Gamma}(\mathbf{x}_i) + b\right] \geq 1, i = 1, \ldots, N. \tag{16}$$

Given the training sets in (7) the binary support vector machine classifier is defined as

$$y(\mathbf{x}) = sign\left[\sum_{i=1}^{N} \alpha_i y_i \Bbbk(\mathbf{x}, \mathbf{x}_i) + b\right]. \qquad (17)$$

The non-zero $\alpha_i's$ are "support values" and the corresponding data points, $\mathbf{x}_i$, are the "support vectors". Quadratic programming is one method of solving for the $\alpha_i's$ and b in the standard SVM algorithm.

C. Multiclass Classification

For the multiclass problem the machine learning algorithm produces estimates with multiple hyperplane separations. The set of input vectors and training labels is defined as $\{\mathbf{x}_i, y_i^c\}_{i=1,c=1}^{i=n,c=C}$, $\mathbf{x}_i \in \mathbb{R}^n$, $y_i \in \{1, \ldots, G\}$, n is the index of the training pattern and C is the number of classes. There exist many SVM approaches to multiclass classification problem. Two primary multiclass techniques are one-vs-one and one-vs-rest. One-vs-one applies SVMs to selected pairs of classes. For C distinct classes there are $\frac{C(C-1)}{2}$ hyperplanes that separate the classes. The one-vs-rest SVM technique generates C hyperplanes that separate each distinct class from the ensemble of the rest. In this paper we only consider the one-vs-one multiclass SVM.

Platt, et.al., [12] introduced the decision directed acyclic graph (DDAG) and a Vapnik-Chervonenkis (VC) analysis of the margins. The DDAG technique is based on $\frac{C(C-1)}{2}$ classifiers for a C class problem, one node for each pair of classes. In [12] it is proved that maximizing the margins at each node of the DDAG will minimize the generalization error. The performance benefit of the DDAG architecture is realized when the ith classifier is selected at the ith/j^{th} node and the jth class is eliminated. Refer to Figure 1 for a diagram of a four class DDAG.
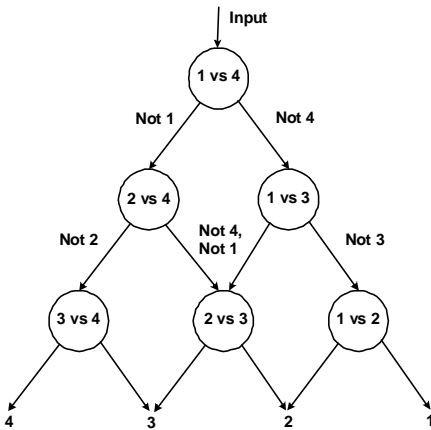
Fig. 1. Four class DDAG for one-vs-one multiclass LS-SVM based DOA estimation.

D. Least Squares SVM

Suykens, et.al., [13] introduced the LS-SVM which is based on the SVM classifier, refer to equation (17). The LS-SVM classifier is generated from the optimization problem:

$$\min_{\mathbf{w},b,\phi} \mathcal{L}_{LS}(\mathbf{w},\phi) = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{2}\gamma\sum_{i=1}^{n}\phi_i^2, \qquad (18)$$

γ and ϕ_i are the regularization and error variables, respectively. The minimization in (18)includes the constraints

$$y_i\left[\mathbf{w}^T\boldsymbol{\Gamma}(\mathbf{x}_i) + b\right] \geq 1 - \phi_i, \quad i = 1, \ldots, n, \qquad (19)$$

The LS-SVM includes one universal parameter, γ, that regulates the complexity of the machine learning model. This parameter is applied to the data in the feature space, the output of the kernel function. A small value of γ minimizes the model complexity, while a large value of γ promotes exact fitting to the training points. The error variable ϕ_i allows misclassifications for overlapping distributions [14].

The Lagrangian of equation (18) is defined as

$$\begin{aligned}\mathcal{Z}_{LS}(\mathbf{w},b,\phi,\alpha) &= \mathcal{L}_{LS}(\mathbf{w},b,\phi) - \qquad (20)\\ &\sum_{i=1}^{n}\alpha_i\left\{y_i\left[\mathbf{w}^T\boldsymbol{\Gamma}(\mathbf{x}_i)+b\right] - 1 + \phi_i\right\}\end{aligned}$$

where α_i are Lagrangian multipliers that can either be positive or negative. The conditions of optimality are

$$\frac{d\mathcal{Z}_{LS}}{d\mathbf{w}} = 0, \; \mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i\boldsymbol{\Gamma}(\mathbf{x}_i) \qquad (21)$$

$$\frac{d\mathcal{Z}_{LS}}{db} = 0, \; \sum_{i=1}^{n}\alpha_i y_i = 0 \qquad (22)$$

$$\frac{d\mathcal{Z}_{LS}}{d\phi} = 0, \; \alpha_i = \gamma\phi_i \qquad (23)$$

$$\frac{d\mathcal{Z}_{LS}}{d\alpha_i} = 0, \; y_i\left[\mathbf{w}^T\boldsymbol{\Gamma}(\mathbf{x}_i)+b\right] - 1 + \phi_i = 0 \qquad (24)$$

A linear system can be constructed from equations $(21) - (24)$ [13],

$$\begin{bmatrix} I & 0 & 0 & -Z^T \\ 0 & 0 & 0 & -Y^T \\ 0 & 0 & \gamma I & -I \\ Z & Y & I & 0 \end{bmatrix}\begin{bmatrix} \mathbf{w} \\ b \\ \phi \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \overrightarrow{1} \end{bmatrix} \qquad (25)$$

$$Z = \left[\boldsymbol{\Gamma}(\mathbf{x}_1)^T y_1, \ldots, \boldsymbol{\Gamma}(\mathbf{x}_n)^T y_n\right] \qquad (26)$$

$$Y = [y_1, \ldots, y_n], \; \overrightarrow{1} = [1, \ldots, 1] \qquad (27)$$

$$\phi = [\phi_1, \ldots, \phi_n], \; \alpha = [\alpha_1, \ldots, \alpha_n] \qquad (28)$$

By eliminating weight vector $\mathbf{w}$ and the error variable ϕ, the linear system is reduced to:

$$\begin{bmatrix} 0 & Y^T \\ Y & ZZ^T + \gamma^{-1}I \end{bmatrix}\begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ \overrightarrow{1} \end{bmatrix} \qquad (29)$$

In the linear systems defined in $(25)-(29)$ the support values α_i are proportional to the errors at the data points. In the standard SVM case many of these support values are zero, but most of the least squares support values are non-zero. In [13] a conjugate gradient method is proposed for finding b and α, which are required for the SVM classifier in equation (17).

IV. ALGORITHMS FOR DOA ESTIMATION

Two primary, classic methods for subspace based DOA estimation exist in literature, Multiple Signal Classification (MUSIC) [15] and Estimation of Signal Parameters Via Rotational Invariance Techniques (ESPRIT) [16]. The MUSIC algorithm is based on the noise subspace and ESPRIT is based on the signal subspace.

Many computational techniques exist for working through limitations of DOA estimation techniques, but currently no techniques exist for a system level approach to accurately estimating the DOAs at the base station. A number of limitations relating to popular DOA estimation techniques are: 1) the signal subspace dimension is not known, many papers assume that it is. The differences between the covariance matrix and the sample covariance matrix add to the uncertainty, 2) searching all possible angles to determine the maximum response of the MUSIC algorithm, 3) evaluating the Root-MUSIC polynomial on the unit circle, 4) multiple eigen decompositions for ESPRIT, 5) computational complexity for maximum likelihood method. The capabilities, in terms of resolution and computational requirements, of these standard DOA estimation algorithms serve as the benchmark for the machine learning based DOA estimation. Refer to Section VII for a comparison between standard DOA estimation algorithms and the one-vs-one multiclass LS-SVM DOA estimation algorithm.

A. Machine Learning for DOA Estimation

To estimate the antenna array response, $\mathbf{z}_j = \sum_{l=1}^{L} \mathbf{a}(\theta_l) q_j^l$, we must know $\mathbf{a}(\theta_l)$ and q_i^l. The continuous pilot signal, included in cdma2000, can be used in estimating q_i^l. This must be done for each resolvable path, i.e., $q_i = \begin{bmatrix} q_i^1, & q_i^2, & \dots & ,q_i^L \end{bmatrix}$. Estimating $\mathbf{A}(\theta) = \begin{bmatrix} \mathbf{a}(\theta_1), & \mathbf{a}(\theta_2), & \dots & ,\mathbf{a}(\theta_L) \end{bmatrix}$ requires information on the DOA.

The process of DOA estimation is to monitor the outputs of D antenna elements and predict the angle of arrival of L signals, $L<D$. The output matrix from the antenna elements is

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}(\theta_1) & \mathbf{a}(\theta_2) & \dots & \mathbf{a}(\theta_L) \end{bmatrix} \tag{30}$$
$$\mathbf{a}(\theta_l) = \begin{bmatrix} 1 & e^{-jk_l} & e^{-j2k_l} & \dots & e^{-j(D-1)k_l} \end{bmatrix}^T,$$

and the vector of incident signals is $\boldsymbol{\theta}_r = \begin{bmatrix} \theta_1, & \theta_2, & \dots & ,\theta_L \end{bmatrix}$. With a training process,

the learning algorithms generate DOA estimates, $\widehat{\boldsymbol{\theta}}_r = \begin{bmatrix} \widehat{\theta}_1, & \widehat{\theta}_2, & \dots & ,\widehat{\theta}_L \end{bmatrix}$, based on the responses from the antenna elements, $\mathbf{a}(\theta_l)$.

For the proposed machine learning technique there is a trade-off between the accuracy of the DOA estimation and antenna array beamwidth. An increase in DOA estimation accuracy translates into a smaller beamwidth and a reduction in MAI. Therefore the accuracy in DOA estimation directly influences the minimum required power transmitted by the mobile. There should be a balance between computing effort and reduction in MAI.

V. LS-SVM DDAG BASED DOA ESTIMATION ALGORITHM

In this paper we propose a multiclass SVM algorithm trained with projection vectors generated from the signal subspace eigenvectors and the sample covariance matrix. The output labels from the SVM system are the DOA estimates.

The one-vs-one multiclass LS-SVM DDAG technique for DOA estimation is trained for C DOA classes. The DDAG tree is initialized with $\frac{C(C-1)}{2}$ nodes. Therefore $\frac{C(C-1)}{2}$ one-vs-one LS-SVMs are trained to generated the hyperplanes with maximum margin. For each class the training vectors, $\mathbf{x}_n$, are generated from the eigenvectors spanning the signal subspace. The number of classes is dependent upon on the antenna sectoring and required resolution. For a CDMA system the desired interference suppression dictates the fixed beamwidth. CDMA offers this flexibility since the all mobiles use the same carrier frequency. For FDMA systems a narrow beamwidth is desired, since frequency reuse determines the capacity of a cellular system.

The signal subspace eigenvectors of the received signal covariance matrix are required for accurate DOA estimation. For a CDMA system with adaptive antenna arrays the covariance matrix of the received signal is

$$\mathbf{R}_{rr} = \mathbb{E}\left[\mathbf{r}_A \mathbf{r}_A^H\right]. \tag{31}$$

In our machine learning based DOA estimation algorithm the principal eigenvectors must be calculated. Eigen decomposition (ED) is the standard computational approach for calculating the eigenvalues and eigenvectors of a the covariance matrix. ED is a computationally intense technique, faster algorithms such as PASTd [17] have been developed for real-time processing applications.

For the LS-SVM based approach to DOA estimation the output of the receiver is used to calculate the sample covariance matrix $\widehat{\mathbf{R}}_{rr}$ of the input data signal $\mathbf{r}_A(k)$,

$$\widehat{\mathbf{R}}_{rr} = \frac{1}{M} \sum_{k=K-M+1}^{K} \mathbf{r}_A(k)\mathbf{r}_A^H(k). \tag{32}$$

The dimension of the observation matrix is $D \times M$, M is ideal sample size (window length), and the dimension of the

TABLE I

PROJECTION COEFFICIENTS FOR MACHINE LEARNING BASED POWER CONTROL

	Projection Coefficients		
	$25°$	$30°$	$40°$
1	0.17+i·0.86	-0.20-i·0.54	0.00+i·0.86
2	0.66+i·0.05	-0.82-i·0.14	0.73-i·0.55
3	0.04-i·0.73	0.28+i·0.96	-1.01-i·0.58
4	-1.08-i·0.50	1.04-i·0.37	0.06+i·1.05
5	-0.60+i·0.92	-0.56-i·1.01	0.72-i·0.61
6	0.60+i·0.74	-0.87+i·0.64	-0.92-i·0.51
7	0.72-i·0.56	0.63+i·0.62	-0.03+i·0.76
8	-0.52-i·0.78	0.51-i·0.44	0.45-i·0.42

sample covariance matrix is $D \times D$. The principal eigenvectors, $\mathbf{v}_1, \ldots, \mathbf{v}_D$, are calculated via eigen decomposition (ED) or subspace tracking techniques. Each eigenvector is used to calculate a covariance matrix, $\widehat{\mathbf{R}}_{vv_1}, \ldots, \widehat{\mathbf{R}}_{vv_D}$.

The algorithm requires only the set of estimated eigenvectors from the sample covariance matrix, which are used to generate projection coefficients for the classification process. The projection vectors are generated from the projection of $\widehat{\mathbf{R}}_{vv_d}$, $1 \leq d \leq D$, onto the primary eigenvector of the signal subspace. In the training phase the hyperplanes at each DDAG node are constructed with these projection vectors. In the testing phase $\widehat{\mathbf{R}}_{vv_d}$ is generated from the received signal $\mathbf{r}_A(k)$ and the principal eigenvectors. Then the projection coefficients for the ith/j^{th} node of the DDAG are computed with dot products of $\widehat{\mathbf{R}}_{vv_d}$ and the ith/j^{th} training eigenvectors. This new set of projection vectors is testing with the ith/j^{th} hyperplane generated during the training phase. The DOA labels are then assigned based on the DDAG evaluation path. A similar projection coefficient technique has been successfully applied to a multiclass SVM facial recognition problem presented in [18]. Table I includes three sets of projection vectors, each set corresponds to a different DOA. From a review of the data it is evident that the classes are not linearly separable. The data must be projected to a higher dimension feature space and tested against the separating hyperplane.

The following algorithm for the one-vs-one multiclass LS-SVM implementation for DOA estimation includes preprocessing, training, and testing steps. Specifically, the algorithm requires two sets of projection vectors for each DDAG node. This allows for automatic MSE calculations at each step of the DDAG evaluation path, thus providing a unique method for error control and validation.

- Preprocessing for SVM Training
  1) Generate the $D \times N$ training signal vectors for the C LS-SVM classes, D is the number of antenna elements, N is the number of samples.
  2) Generate the C sample covariance matrices, $\mathbf{U}$, with M samples from the $D \times N$ data vector.
  3) Calculate the signal eigenvector, $\mathbf{S}$, from each of the C sample covariance matrices.
  4) Calculate the $D \times 1$ projection vectors, $\mathbf{U} \cdot \mathbf{S}$, for each of the C classes. The ensemble of projection vectors consists of $\frac{N}{M}$ samples.
  5) Store the projection vectors for the training phase and the eigenvectors for the testing phase.
- LS-SVM Training
  1) With the C projection vectors train the $\frac{C(C-1)}{2}$ nodes with the one-vs-one LS-SVM algorithm.
  2) Store the LS-SVM variables, α_i and b from equation (17), which define the hyperplane separation for each DDAG node.
- Preprocessing for SVM Testing
  1) Acquire $D \times N$ input signal from antenna array, this signal has unknown DOAs.
  2) Generate the sample covariance matrix with M samples from the $D \times N$ data vector.
  3) Calculate the eigenvectors for the signal subspace and the noise subspace.
  4) Generate the covariance matrices for each eigenvector.
- LS-SVM Testing for the i/j DDAG Node
  1) Calculate **TWO** $D \times 1$ projection vectors with the desired eigenvector covariance matrix and the ith and j^{th} eigenvectors from the training phase.
  2) Test both projection vectors against the LS-SVM hyperplane for the i/j node. This requires two separate LS-SVM testing cycles, one with the projection vector from the ith eigenvector and one with the projection vector from the jth eigenvector.
  3) Calculate the mean value of the two LS-SVM output vectors (labels). Select the mean value that is closest to a decision boundary, 0 or 1. Compare this value to the label definition at the node, then select the proper label.
  4) Repeat process for the next DDAG node in the evaluation path or declare the final DOA label.
- Error Control
  1) Review the MSE calculations for the DDAG evaluation path.
  2) Apply error control and validation measures to classify the label as either an accurate **DOA** estimate or as **NOISE**.

VI. SIMULATION RESULTS

Two simulation plots are included below. Each simulation consists of a four class LS-SVM DDAG system. Figure 2 shows results for a ten degree range per class. Figure 3 shows results for a one degree range per class.

The antenna array includes eight elements, therefore the training and test signals were 8×1 vectors. The training

and test signals are the complex outputs from the antenna array. The received complex signal is modeled with a zero mean normal distribution with unit variance; the additive noise includes a zero mean distribution with a 0.2 variance. This combination of signal and noise power translates into a $7dB\ SIR$.

The system training consists of six DDAG nodes for the four DOA classes. Both the training and test signals consisted of 1500 samples and the window length of the sample covariance matrix was set to five. Therefore the training and test sets were composed of 300 samples of each 8×1 projection vector.

To completely test the LS-SVM DDAG system's capabilities the simulation were automated to test a wide range of DOAs. The DOA test set consisting of signals ranging from three degrees before the first DOA class to three degrees after the last DOA class. Thus there were forty-six test signals for Figure 2 and fourteen test signals for Figure 3. As can been seen from the two plots the LS-SVM DDAG DOA estimation algorithm is extremely accurate. No misclassifications were logged. Testing shows that the LS-SVM DDAG system accurately classifies the DOAs for any desired number of classes and DOA separations from one degree to twenty degrees.
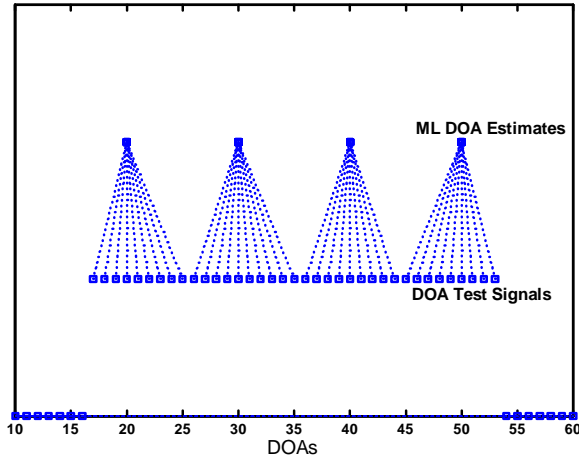
Fig. 2.   LS-SVM for DOA estimation, four classes with ten degree separation between each.

A. Decision Grids

The decision grid (DG) technique was developed to track the DDAG evaluation path and generate statistics to characterize the confidence level of the DOA classifications. The theoretical DG (T-DG) is a technique we developed to quantify errors and add insight into the robustness of the LS-SVM DDAG architecture. The T-DG is a deterministic $2D$ grid for DDAGs with a relatively small number of classes and small DOA range between classes. The elements of the T-DG
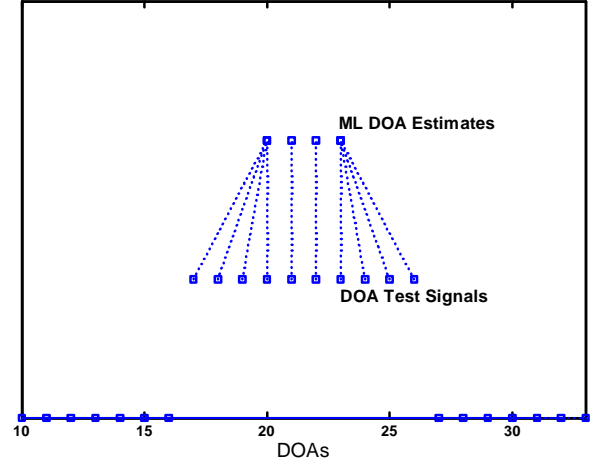
Fig. 3.   LS-SVM for DOA estimation, four classes with one degree separation between each.

represent the deterministic values of the two LS-SVM labels at each DDAG level, the deterministic values are referred to as "theoretical decision statistics". Designing T-DGs for DDAGs with three to five classes and DOA ranges up to five degrees between classes is straight forward. The T-DGs are not deterministic for large DOA ranges, i.e. for a DOA range of ten degrees between classes empirical results show that the DDAG evaluation path is unpredictable. The large DOA ranges lead to uncertainty in the evaluation path, even though the test DOA is classified correctly.

Empirical decision grids (E-DG) are automatically generated in the LS-SVM DDAG DOA estimation algorithm. The E-DGs tabulate the mean of the LS-SVM output label vectors at each DDAG node and level, the mean values are referred to as "decision statistics". The unique design of this algorithm includes testing the input data against two hyperplanes at the ith/j^{th} node. With this approach the two output vectors at each node are compared to one another. In a noise-free environment, with perfect classification, the two label vectors would be binary opposites, i.e. one label vector would be all $0's$ and the other label vector would be all $1's$. This technique enables computation of theoretical mean square errors and empirical mean square errors, refer to Section VI-B.

Table II includes a standard T-DG and Tables III and IV include E-DGs for a three class DDAG with a two degree DOA range per class. The two levels of a three class DDAG are equivalent to the first two levels of a four class DDAG, refer to Figure 1. Table II includes the possible evaluation paths of this three class DDAG. The nodes for each DOA evaluation path are included for the first and second DDAG level. For example, DOA 1 has an evaluation path of Node 1 vs 3 at Level 1 and Node 1 vs 2 at Level 2. In Table III E-DG presents the decision statistics for a signal subspace eigenvector; in Table IV the second E-DG presents the

TABLE II

THEORETIC DECISION GRID FOR A DDAG SYSTEM WITH 3 CLASSES AND A 2 DEGREE DOA RANGE.

	DOAs				
	Class 1		Class 2		Class 3
T-DG, Level 1	1	2	3	4	5
Node	1vs3	1vs3	1vs3	1vs3	1vs3
Label 0	0	0	0.5	1	1
Label 1	1	1	0.5	0	0
T-DG, Level 2					
Node	1vs2	1vs2	1vs2\2vs3	2vs3	2vs3
Label 0	0	0.5	0\1	0.5	1
Label 1	1	0.5	1\0	0.5	0

TABLE III

EMPIRICAL DECISION GRID FOR A SIGNAL EIGENVECTOR

	DOAs				
Signal Data	Class 1		Class 2		Class 3
E-DG, Level 1	1	2	3	4	5
Node	1vs3	1vs3	1vs3	1vs3	1vs3
Label 0	0	0	0.032	0.952	1
Label 1	1	1	0.576	0	0
E-DG, Level 2					
Node	1vs2	1vs2	1vs2\2vs3	2vs3	2vs3
Label 0	0	0.176	1	0.808	1
Label 1	1	0.816	0	0.496	0

TABLE IV

EMPIRICAL DECISION GRID FOR A NOISE EIGENVECTOR

	DOAs				
Noise Data	Class 1		Class 2		Class 3
E-DG, Level 1	1	2	3	4	5
Node	1vs3	1vs3	1vs3	1vs3	1vs3
Label 0	0.328	0.376	0.304	0.352	0.384
Label 1	0.752	0.744	0.712	0.768	0.776
E-DG, Level 2					
Node	1vs2	1vs2	1vs2\2vs3	2vs3	2vs3
Label 0	0.232	0.256	0.144	0.136	0.184
Label 1	0.896	0.904	0.952	0.944	0.944

decision statistics for a noise subspace eigenvector.

B. Theoretical and Empirical MSEs

The difficulty in tracking the performance of the LS-SVM DDAG DOA estimation algorithm is due to the numerous DDAG evaluation paths. For many DDAGs the evaluation paths can be determined based on the input data and the class definitions. How can decision statistics be applied to performance characterization?

The two primary performance measures for the LS-SVM DDAG are the theoretical MSE (T-MSE) and the empirical MSE (E-MSE). Both MSE performance measures are based on MSE calculations with T-DGs and E-DGs. The T-MSE is a MSE calculation between the corresponding elements of the T-DG and the E-DG. This is a measure of the algorithm's

empirical decision statistics in relation to the "theoretical" decision statistics. For example, the T-MSE for a 3 class DDAG is calculated with the T-DG and E-DG presented in Tables II and III. The T-MSE for Class 2 is calculated as

Level 1		Level 2	
Label 0	Label 1	Label 0	Label 1
$(0.5 - 0.032)^2$	$(0.5 - 0.576)^2$	$(1 - 1)^2$	$(0 - 0)^2$

Unlike the T-MSE, the E-MSE is a technique that allows for real-time error tracking with only the empirical decision statistics. The E-MSE uses only the E-DGs and the differences between the two LS-SVM decision statistics at each node in the evaluation path. This is a measure of the empirical classification accuracy achieved at each DDAG node. The E-MSE for a 3 class DDAG is calculated with only the E-DG presented in Table III. The MSE for Class 2, Level 1 is $(|0.032 - 0.576| - 1)^2 = 0.208$ and the MSE for Class 2, Level 2 is $(|1 - 0| - 1)^2 = 0$.

C. Misclassifications vs. Gross Errors

Two secondary performance measures for the LS-SVM DDAG are misclassifications and gross errors. These measures are used for performance characterization of the multiclass LS-SVM DDAG DOA estimation algorithm and for tracking variations in performance for various algorithm parameters. Misclassifications and gross errors can not be used in real time implementation because knowledge of the test DOAs is required.

Misclassifications measure "small shifts" in DOA classifications. If a DOA is located near a border between labels the machine learning process could classify the data to an adjacent label, not the closest label. Therefore, a misclassification is a shift related error where a signal is detected, but classified to a spatially adjacent label. This type of error still gives an indication of the received DOA. The region of misclassifications is defined as $\frac{1}{2}$ of the DOA range applied to both sides of a DOA class.

Gross errors measure significant errors in DOA classifications. If a DOA is classified into a specific class, but spatially located at least one entire class away, then the error is due to a breakdown in the machine learning process. This type of error assigns false/misleading information to a received DOA. The region of gross errors is defined as the magnitude of the DOA range applied to both sides of the DOA class.

Figure 4 displays the DOA regions for correct classifications, misclassifications and gross errors. This specific example is for a DDAG class centered at $0°$ with a $5°$ DOA range, i.e., any DOA in the range $[-2, 2]$ is correctly classified to the $0°$ class. The region enclosed by the dashed brackets includes all DOAs that are correctly classified at the $0°$ class. If any DOAs outside the dashed brackets but inside the solid brackets are assigned the $0°$ class, then that DOA would be a misclassification. If any DOAs outside the solid brackets are assigned to the $0°$ class, then that DOA

would be a gross error. The misclassification region, for a DOA classified at $0°$, is $DOA \in [-4, -3], [3, 4]$. The gross error region, for a DOA classified at $0°$, is $DOA \notin [-4, 4]$.
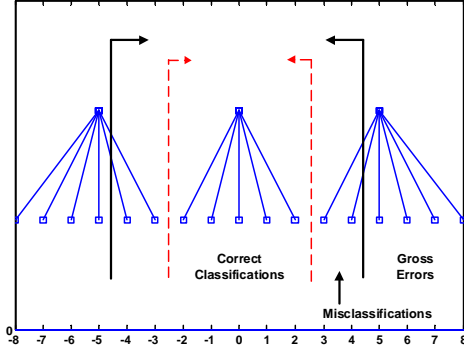
Fig. 4. Diagram of regions defining DOA misclassifications and gross errors.

Fig. 5. Theoretical MSE for a the polynomial kernel, the DOA range is $5°$ and spans the DDAG classes at $30°$, $35°$, $40°$, $45°$, the LS-SVM parameter, γ, is set at 2.

D. Kernel Parameters

Simulation results show that kernel selection has the greatest effect, out of all tunable variables, in the classification process. The four kernels discussed in Section III-A are tested with the LS-SVM DDAG DOA estimation algorithm. The performances of each kernel function and the associated parameters are characterized with in terms of MSE, misclassifications, and gross errors. In addition, the LS-SVM regularization parameter, γ, is varied to show the influence of the LS-SVM complexity.

1) Polynomial Kernel: The polynomial kernel provides the best results, in relation to the RBF, MLP, and linear kernels. Figure 5 displays the T-MSE in terms of the polynomial degree, $\mathbf{d}$, and constant, $\boldsymbol{\theta}$. The simulation is based on a four class DDAG with a $5°$ DOA range and a fixed LS-SVM variable, $\gamma = 2$. The results show that the degree of the polynomial kernel affects the DOA estimation; the best values are $\mathbf{d} = 2$ and $\mathbf{d} = 4$. For $\mathbf{d} = 1$ the polynomial kernel is equivalent to the linear kernel. The MSE is constant for $1 \leq \gamma \leq 6$, and the polynomial constant, $\boldsymbol{\theta}$, does not influence the performance. The rate of misclassifications is 1.2% with zero gross errors. The degree of the polynomial is the only factor affecting the computational time for system training.

2) Radial Basis Function Kernel: The performance of the RBF kernel is characterized in terms of the LS-SVM regularization variable, γ, and the smoothing parameter, σ^2. The simulation is based on a four class DDAG with a $5°$ DOA range. The results show that the MSE is constant for $\gamma \geq 1.5$, and $\sigma^2 \geq 0.5$. The rate of misclassifications is 0.4% with zero gross errors. The training time increases with the value of γ and for small values of σ^2. The performance of the RBF kernel matches the performance of the polynomial kernel for DOAs in the range of $15°$ to $60°$. The performance of the polynomial kernel exceeds that of the RBF kernel for DOAs $< 15°$ and $> 60°$.

3) Multilayer Perceptron Kernel: Results show that the MLP kernel is ineffective in maintaining a low MSE for the range of parameters tested. The rate of misclassifications is 42.5% and the rate of gross errors is 17.2%. Overall the performance of the MLP kernel is inferior to the polynomial and RBF kernels.

4) Linear Kernel: The linear kernel is equivalent to the polynomial kernel with $\mathbf{d} = \mathbf{1}$. Large MSE values show that the linear kernel is not effective in the LS-SVM DOA estimation algorithm. The average T-MSE is 27.8% and the average E-MSE is 61.1%.

E. Training and Test Vectors

The design of training sequences is an important factor in machine learning applications. For adaptive antenna arrays the training sequences represent the array outputs for the C DOA classes. Three specific elements of the training sequences are noise variance, training vector length, and length of the sample covariance window. The requirement is to design training sequences that minimize both the training error and generalization error. Empirical analysis of the multiclass LS-SVM based DOA estimation algorithm shows that training error is effectively zero; the hyperplane separation of the data in the feature space is well defined and separable. In this paper the generalization error is expressed in terms of MSEs, misclassifications and gross errors.

The primary method for training LS-SVM DDAG systems for DOA estimation is based on synthetic training vectors generated with known noise power and preselected vector lengths. In practice, the training vectors would be stored in the memory of the receiver that employs the DOA estimation

algorithm. This approach allows for offline training of the binary LS-SVM algorithms.

Simulation results show that the LS-SVM DOA estimation algorithm is robust, in terms of MSE, when analyzed for a range of SIRs in the training vectors and the test signals. In general, the noise power of the training vectors doesn't have a dramatic effect on the generalization error. Simulations were conducted with training vectors that included SIRs in the range of 20 dB to 7 dB. Review of the misclassification and gross error statistics show that training vectors with noise variances of 0.04 and 0.12, which correspond to SIRs of 13 dB and 10 dB, provide the best performance.

1) Length of Training and Testing Vectors: Figure 6 includes two plots of average theoretical MSE versus training vector length. The data is specific to a four class LS-SVM DDAG system with a four degree polynomial kernel. The two plots show that the window length of the sample covariance matrix does not impact the performance. Likewise there is no correlation between the length of the training vector and the MSE. The results in Figure 6 are based on test vectors with size equivalent to the training vectors. Figure 7 is a 3D plot of the theoretical MSE as a function of vector dimensions; the dimensions of the training vectors and input data vectors. The length of the input data vector ranges from 0.5 to 2 times the length of the training vectors. The data shows that range of input data vectors has no effect on the MSE statistics.

Fig. 7. Theoretical MSE as a function of training vector length and input vector length. The LS-SVM DDAG system includes four class and a four degree polynomial kernel. The test window multiplier defines the input vector length, i.e. the input vector length ranges between 0.5 to 2 times the training vector length.

TABLE V

PROCESSING TIMES, IN SECONDS, FOR ONE-VS-ONE MULTICLASS LS-SVM FOR DOA ESTIMATION.

	Vector Size							
	25	50	75	100	125	150	175	200
Train	0.30	0.94	2.25	4.49	7.39	11.27	15.23	20.38
Test	0.20	0.23	0.31	0.47	0.56	0.66	0.72	0.91

Fig. 6. Average theoretical MSE as a function of training vector length. Two data plots are included; one plot is for a sample covariance matrix with a five sample window, one plot is for a sample covariance matrix with a ten sample window.

Table V shows the processing times, in seconds, required for training a four class LS-SVM DDAG system with a four degree polynomial kernel. and testing the input data. The results Data is included for training and test vectors that range from 25 samples to 200 samples. The simulations were conducted with a Pentium 4 running at 2.5 GHz. The processing times are relative to the computer system and the level of optimization applied to the programming, but serve

as a basic indicator for possible hardware implementation and real-time applications.

The data in this section shows that the design of the training vectors is important, but there is a tolerance in the selection of noise power and training vector length. The available tolerance in choosing parameters of the training vectors validates the design of the LS-SVM DOA estimation algorithm. This characteristic allows flexibility in the system design and provides a high confidence level in the DOA estimates. In addition, when considering real-time implementation of the algorithm, the dimensions of the training vector must be carefully reviewed. Shorter training vectors offer high performance, in terms of MSE, and fast training times.

F. Range of DDAG Parameters for DOA Estimation

The exceptional performance of the LS-SVM DDAG DOA estimation algorithm has been proved in the previous sections. Most the previous simulation results were based on three and four class DDAGs. To cover the desired span of the antenna array sector the algorithm must be flexible in the number of DDAG classes and DOA ranges. Different applications require different DDAG architectures. Many times the application will require fast training and high accuracy. Training a LS-SVM DDAG system can be performed offline. But covering a large antenna sector with high resolution would require either:

TABLE VI
PERCENTAGE OF MISCLASSIFICATIONS VERSUS DDAG CLASSES (3-6)
AND DOA RANGES (1-10).

Classes	DOA Range between Classes, Degrees									
	1	2	3	4	5	6	7	8	9	10
3	0	0	0	0	6.7	0	4.8	4.2	0	0
4	0	0	0	0	0	0	3.6	3.1	0	0
5	0	0	0	0	4.0	0	2.9	0	6.7	0
6	0	0	0	0	0	0	4.8	0	5.6	0

1) A DDAG with a large number of classes and a small DOA range,
2) A two stage system where the antenna sector is partitioned into a set number of classes with a wide DOA range. First, the signal is detected in a specific partition, then a DDAG structure for high resolution can classify the DOA with high accuracy

Whatever the desired approach is, the LS-SVM DDAG algorithm must be flexible in design and robust in performance.

The data in this section proves the performance for a wide range of DDAG structures. Simulations were conducted for three to ten classes with DOA ranges between $1°$ and $20°$. With these classes and DOA ranges the LS-SVM DDAG algorithms is able to span antenna sectors of $3°$ to $90°$. Table VI lists the number of misclassifications. Seventy-five percent of the DDAG structures with DOA ranges between $1°$ and $10°$ have zero misclassifications; the average rate of misclassifications for the set of DDAG structures is 1.2%. The largest percentage of misclassifications is 6.7% and occurs with a five class DDAG with a nine degree DOA range.

G. Multilabel Capability for Multiple DOAs

In DOA estimation for cellular systems, there can be multiple DOAs for a given signal. This results from multipath effects induced by the communication channel. The machine learning system must be able to discriminate between a small number of independent DOAs that include signal components with similar time delays. With this constraint the machine learning algorithm then must be a multiclass system and able to process multiple labels.

The machine learning algorithm must generate multiclass labels, $y_i \, \epsilon \, C$, where $C \, \epsilon \, [-90, 90]$ is a set of real numbers that represent an appropriate range of expected DOA values, and multiple labels $y_i, i = 1 \ldots L$ for L dominant signal paths. If antenna sectoring is used in the cellular system the multiclass labels are from the set $C \, \epsilon \, [\mathbb{S}_i]$, where $\mathbb{S}_i$ is field of view for the ith sector.

Multilabel classification is possible with the LS-SVM DDAG algorithm presented in Section V. The machine learning algorithm for DOA estimation assigns DOA labels to each eigenvector in the signal subspace. By repeating the

DDAG cycle for each eigenvector the multiclass algorithm has the capability of assigning multiple labels to the input signal.

VII. COMPARISON TO STANDARD DOA ESTIMATION ALGORITHMS

The performance of the one-vs-one multiclass LS-SVM algorithm for DOE estimation is described, in detail, in the previous section. The results show that the multiclass classification approach to DOA estimation provides unique benefits, in terms of computational complexity and flexibility. Each algorithm is trained for C DOA classes. The number of classes is dependent upon on the antenna sectoring and required resolution. The ideal application of this technique is CDMA cellular systems. For a CDMA system the desired interference suppression dictates the fixed beamwidth. A reduction in beamwidth corresponds to a reduction in MAI, thus reducing the required transmit power at the mobile subscriber. CDMA offers this flexibility since the all mobiles use the same carrier frequency. For Frequency Division Multiple Access (FDMA) systems a narrow beamwidth is desired, since frequency reuse factors into the capacity of a cellular system, thus requiring accurate DOA estimates with high resolution.

A. Computational Complexity

Conventional subspace based DOA estimation algorithms, such as MUSIC and ESPRIT, are computationally complex. The algorithms require accurate knowledge of the signal subspace dimension and accurate estimates of the signal and noise subspace eigenvectors. Additionally, the MUSIC algorithm requires a precise characterization of the antenna array and the ESPRIT algorithm requires multiple eigen decompositions.

The one-vs-one multiclass LS-SVM algorithm for DOA estimation is flexible, with respect to computationally requirements. The training cycle for the LS-SVM based DDAG is straight forward and can be completed offline with simulated data. The only information required is the size of the antenna array and the number of DDAG nodes, which corresponds to DOA classes. For accurate DOA estimates the only information required, for the LS-SVM DDAG testing cycle, is the dimension of the antenna array and accurate eigenvector estimates of the sample covariance matrix. The dimension of the signal subspace is not required, nor is accurate characterization of the antenna array.

B. Simulation Results

Figure 8 compares the one-vs-one multiclass LS-SVM DOA estimation algorithm and the MUSIC algorithm. The top window shows perfect DOA estimation for the machine learning method presented in this paper. The multiclass

algorithm includes an eight class DDAG and a one degree DOA range per class. Note that multiclass LS-SVM algorithm classifies signals outside the DOA classes to the nearest class, as shown with the DOAs at $12° - 14°$ and $23° - 25°$. The bottom window displays the DOA estimation with the MUSIC algorithm, 100 DOA estimates are averaged for each received signal and the amplitudes are normalized to the largest estimate. The plots show that the resolution capabilities one-vs-one multiclass LS-SVM DOA estimation algorithm equal that of the MUSIC algorithm.One drawback of the MUSIC algorithm is the broad width of the DOA estimate; a level detection step is required to accurately select the maximum response.

Figure 9 compares the errors and DOA estimates of each algorithm. For this simulation the one-vs-one multiclass LS-SVM algorithm includes a seventeen class DDAG and a five degree DOA range per class. The top window plots the errors in the DOA estimates for ninety degree antenna sector and one DOA sample per degree. The definitions of an error are specific to the two algorithms. For the machine learning based algorithm, an error is defined as a DOA that is classified into a wrong DOA class. For the MUSIC algorithm an error is the difference between the estimated DOA and the actual DOA. As shown in the top window, the only errors associated with the LS-SVM based algorithm occur for DOAs greater than $82°$. The DOAs in error are classified into the spatially adjacent DOA class at $80°$. Likewise, the errors associated with the MUSIC algorithm, that are greater than $1°$, occur for DOAs greater than $70°$. The plots in Figure 9 prove the robust performance of the one-vs-one multiclass LS-SVM algorithm for DOA estimation.
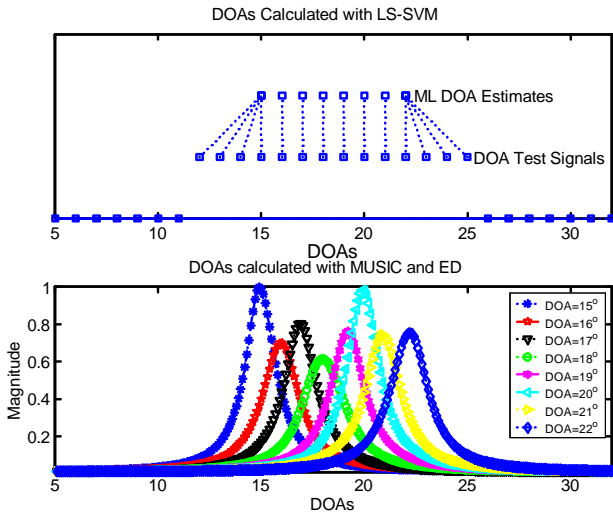
Fig. 8. Comparision between the LS-SVM based DOA estimation algorithm and the MUSIC algorithm. The one-vs-one multiclass LS-SVM DOA estimation algorithm includes eight classes and a one degree DOA range.
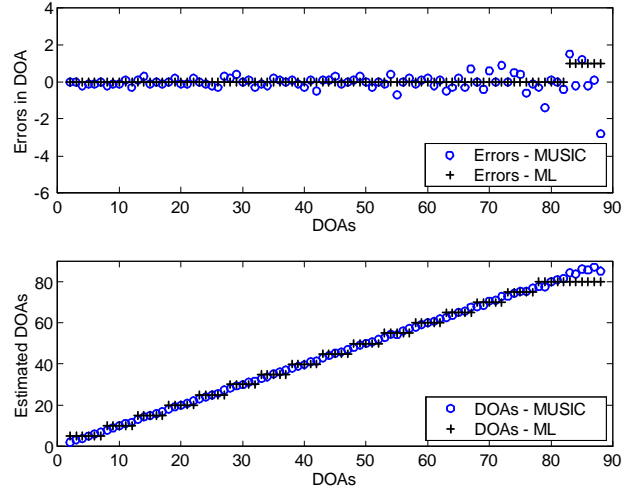
Fig. 9. Comparision of errors and estimated DOAs for the LS-SVM based DOA estimation algorithm and the MUSIC algorithm. The one-vs-one multiclass LS-SVM DOA estimation algorithm includes seventeen classes and a five degree DOA range.

C. Benefits over Standard Techniques

Evaluation of the performance statistics, Section VI, proves that the one-vs-one multiclass LS-SVM algorithm for DOA estimation is reliable with a high degree of accuracy. In terms of performance our new algorithm provides the same capabilities as the standard DOA estimation methods. Specifically, accurate DOA estimates, to a one degree resolution, can be achieved with the standard subspace based algorithms and our machine learning based algorithm. The primary benefits of our LS-SVM based DOA estimation algorithm are the reduced computational complexity, described above, and the flexibility, in terms of DOA classes versus requirements. The specific application dictates the desired resolution and therefore the number of DOA classes. For example, one application may include a sixty degree antenna sector and a desired resolution of ten degrees. These requirements would translate into a seven class system. Another application may include a twenty degree sector and a desired resolution of two degrees; this would translate into a eleven class system. An additional option is to place two DDAG systems in series, as described in Section VI-F, that allows for a high resolution with a small number of classes. In general, the one-vs-one multiclass LS-SVM algorithm for DOA estimation can be adapted to specific requirements, as influenced by system capacity, channel conditions, and available computational resources. The MUSIC and ESPRIT algorithms offer no flexibility, in terms of DOA resolution and computational resources.

VIII. CONCLUSION

In this paper we presented a machine learning architecture for DOA estimation as applied to a CDMA cellular system.

The broad range of our research in machine learning based DOA estimation includes multiclass and multilabel classification, classification accuracy, error control and validation, kernel selection, estimation of signal subspace dimension, and overall performance characterization. We presented an overview of a multiclass SVM learning method and successful implementation of a one-vs-one multiclass LS-SVM DDAG system for DOA estimation.

The LS-SVM DOA estimation algorithm is superior to standard techniques due to the robust design that is insensitive to received SIR, Doppler shift, size of the antenna array, and the computational requirements are adaptable to the desired applications. The algorithm was designed with a multiclass, multilabel capability and includes an error control and validation process. In addition, there are many limitations of standard DOA estimation algorithms, ESPRIT and MUSIC, that do not exist with the LS-SVM DOA estimation algorithm.

The LS-SVM algorithm for DOA estimation assigns DOA labels to each eigenvector in the signal subspace. By repeating the DDAG cycle for each eigenvector the multiclass algorithm has the capability of assigning multiple labels to the input signal. Simulation results show a high degree of accuracy and prove that the LS-SVM DDAG system has a wide range of performance capabilities. The results show that the algorithm is accurate for a large range of DDAG performance independent of DDAG class or DOA range per class. The LS-SVM DDAG system accurately classifies the DOAs for three to ten classes and DOA ranges from one degree to twenty degrees.

REFERENCES

[1] J.C. Liberti, Jr. and T.S. Rappaport, *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*, Prentice Hall, Upper Saddle River, NJ, 1999.

[2] J.H. Winters, "Signal Acquisition and Tracking with Adaptive Arrays in the Digital Mobile Radio System IS-54 with Flat Fading," IEEE Transactions on Vehicular Technology, Vol. 42, No. 4, 377-384, November 1993.

[3] Z. Raida, "Steering an Adaptive Antenna Array by the Simplified Kalman Filter," IEEE Transactions on Antennas and Propagation, Vol. 43, No. 6, 627-629, June 1995.

[4] A.H. El Zooghby, C.G. Christodoulou, and M. Georgiopoulos, "A Neural Network-Based Smart Antenna For Multiple Source Tracking", IEEE Transactions On Antennas and Propagation, vol. 48, no. 5, pp. 768-776, May 2000

[5] A.H. El Zooghby, C.G. Christodoulou, and M. Georgiopoulos, "Performance of Radial-Basis Function Networks for Direction of Arrival Estimation with Antenna Arrays", IEEE Transactions On Antennas and Propagation, vol. 45, no. 11, pp. 1611-1617, November 1997

[6] Ahmed H. El Zooghby, Christos G. Christodoulou, and Michael Georgiopoulos, "Neural Network-Based Adaptive Beamforming for One- and Two- Dimensional Antenna Arrays", IEEE Transactions On Antennas and Propagation, vol. 46, no. 12, pp. 1891-1893, December 1998

[7] Richard O. Duda, Perter E. Hart, and David G. Stork, *Pattern Classification, Second Edition* John Wiley & Sons, New York, NY, 2001.

[8] F. Rashid-Farrokhi, L. Tassiulas, K.J. Ray Liu, "Joint Optimum Power Control and Beamforming in Wireless Networks Using Antenna Arrays", IEEE Transactions On Communications, vol. 46, no. 10, pp. 1313-1324, October 1998.

[9] D.J. Sebald and J.A. Bucklew, "Support Vector Machine Techniques for Nonlinear Equalization", IEEE Transactions On Signal Processing, vol. 48, no. 11, pp. 3217-3226, November 2000.

[10] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*, Cambridge University Press, New York, *2000.*

[11] J.A.K. Suykens, "Support Vector Machines: A Nonlinear Modelling and Control Perspective", European Journal of Control, vol 7, pp. 311-327, 2001

[12] J.C. Platt, N.Christianini, and J. Shawe-Taylor, "Large Margin DAGs for Multiclass Classification", in *Advances in Neural Information Processing Systems*, vol. 12, pp. 547-553, Cambridge, MA, MIT Press, 2000.

[13] J.A.K. Suykens, L. Lukas, P. Van Dooren, B. DeMoor, and J. Vandewalle, "Least Squares Support Vector Machine Classifiers: a Large Scale Algorithm", ECCTD'99 European Conf. on Circuit Theory and Design, pp. 839-842, August 1999.

[14] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines* World Scientific, New Jersey, 2002.

[15] R.O. Schmidt, "Multiple Emitter Location and Signal Parameter Estimation", IEEE Transactions on Antennas and Propagation, AP-34, pp. 276-280, March 1986.

[16] R.H. Roy, and T. Kailath, "ESPRIT-Estimation of Signal Parameters Via Rotational Invariance Techniques", IEEE Transactions On Acoustics, Speech, and Signal Processing, vol. 37, no. 7, pp. 984-995, July 1989.

[17] B. Yang, "Projection Approximation Subspace Tracking", IEEE Transactions on Signal Processing, vol. 43, no. 1, pp. 95-107, January 1995.

[18] G. Guo, S.Z. Li, and K.L. Chan, "Support Vector Machines for Face Recognition", Image and Vision Computing, vol 19, pp. 631-638, 2001.

Judd A. Rohwer received his BS degree in electrical engineering in 1996 from GMI Engineering & Management Institute, Flint, MI and his MS and Ph.D. degrees in electrical engineering from University of New Mexico in 2000 and 2003 respectively. Upon his graduation from GMI he accepted a technical staff position at Sandia National Laboratories, where he has been involved with various system engineering and R&D programs. Dr. Rohwer is a Member of the IEEE and his research interests include signal and image processing, wireless communications, wireless networks, and machine learning.
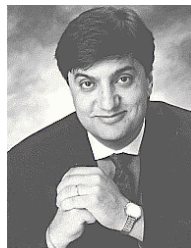
Chaouki T. Abdallah obtained his MS and Ph.D. in Electrical Engineering from GA Tech in 1982, and 1988 respectively. He joined the Electrical and Computer Engineering department at the University of New Mexico where he is currently professor, associate chair, and the director of the graduate program. Dr. Abdallah was Exhibit Chair of the 1990 International Conference on Acoustics, Speech, and Signal Processing (ICASSP), and the Local Arrangements Chair for the 1997 American Control Conference. He is currently serving as the Program Chair for the 2003 Conference on Decision and Control. His research interests are in the area of wireless communications, robust control, and adaptive and nonlinear systems. He is a co-editor of the IEEE Press book Robot Control: Dynamics, Motion Planning, and Analysis; co-author of the book Control of Robot Manipulators, published by Macmillan, and of Linear Quadratic Control: An Introduction, published by Prentice Hall. Dr. Abdallah is a senior member of IEEE and a recipient of the IEEE Millennium medal.

NEURAL NETWORKS FOR THE CALCULATION OF BANDWIDTH OF RECTANGULAR MICROSTRIP ANTENNAS

S. Sinan Gultekin[1], Kerim Guney[2], Seref Sagiroglu[3]

[1]Selcuk University, Faculty of Engineering and Architecture, Department of Electric and Electronic Engineering, 42031, Konya, Turkey, e-mail:sgultekin@selcuk.edu.tr
[2]Erciyes University, Faculty of Engineering, Department of Electronic Engineering, 38039, Kayseri, Turkey, e-mail:kguney@erciyes.edu.tr
[3]Erciyes University, Faculty of Engineering, Department of Computer Engineering, 38039, Kayseri, Turkey, e-mail:ss@erciyes.edu.tr

ABSTRACT: Neural models for calculating the bandwidth of electrically thin and thick rectangular microstrip antennas, based on the multilayered perceptrons and the radial basis function networks, are presented. Thirteen learning algorithms, the conjugate gradient of Fletcher-Reeves, Levenberg-Marquardt, scaled conjugate gradient, resilient backpropagation, conjugate gradient of Powell-Beale, conjugate gradient of Polak-Ribiére, bayesian regularization, one-step secant, backpropagation with adaptive learning rate, Broyden-Fletcher-Goldfarb-Shanno, backpropagation with momentum, directed random search and genetic algorithm, are used to train the multilayered perceptrons. The radial basis function network is trained by the extended delta-bar-delta algorithm. The bandwidth results obtained by using neural models are in very good agreement with the experimental results available in the literature. When the performances of neural models are compared with each other, the best results for training and test were obtained from the multilayered perceptrons trained by the conjugate gradient of Powell-Beale and Broyden-Fletcher-Goldfarb-Shanno algorithms, respectively.

1. INTRODUCTION

Microstrip antennas (MSAs) have become the favorite choice of antenna designers because they offer the attractive features of low profile, light weight, low cost, conformability to curved surfaces, ease of manufacture, and compatibility with integrated circuit technology [1-18]. A number of methods [1-36] using different levels of approximation have been proposed and used to compute the bandwidth of rectangular MSA, as this is one of the most popular and convenient shapes. These methods can generally be divided into two groups: simple analytical methods and rigorous numerical methods. Simple analytical methods can give a good intuitive explanation of antenna radiation properties. However, these methods do not consider rigorously the effects of surface waves. Exact mathematical formulations in rigorous methods involve extensive numerical procedures, resulting in round-off errors, and may also need final experimental adjustments to the theoretical results. These methods also require

high performance large-scale computer resources and a very large number of computations. Furthermore, most of the previous theoretical and experimental work has been carried out only with electrically thin MSAs, normally of the order of $h/\lambda_d \leq 0.02$, where h is the thickness of the dielectric substrate and λ_d is the wavelength in the substrate. Recent interest has developed in radiators etched on electrically thick substrates. The need for theoretical and experimental studies of MSAs with electrically-thick substrates is motivated by several major factors. Among these is the fact that MSAs are currently being considered for use in millimetre-wave systems. The substrates proposed for such applications often have high relative dielectric constants and, hence, appear electrically thick. The need for greater bandwidth is another reason for studying thick substrate MSAs. Consequently, this problem, particularly the bandwidth aspect, has received considerable attention.

In this paper, models based on artificial neural networks (ANNs) are presented for the bandwidth of both electrically thin and thick rectangular MSAs. Ability and adaptability to learn, generalizability, smaller information requirement, fast real-time operation, and ease of implementation features have made ANNs popular in the last few years [37-40]. Because of these fascinating features, artificial neural networks in this article are used to model the relationship between the parameters of MSA and the measured bandwidth results.

In previous works [35,41-48], we also successfully introduced ANNs to compute the various parameters of the triangular, rectangular and circular MSAs. In reference [35], the bandwidth of rectangular MSAs has been computed by using ANNs. In [35], only the multilayered perceptrons (MLPs) were used as the neural network architecture. However, in this paper, both the MLPs and the radial basis function networks (RBFNs) are used for calculating the bandwidth. Furthermore, in [35], the four learning algorithms, the backpropagation (BP) [49], the delta-bar-delta (DBD) [50], the quick propagation (QP) [51], and the extended delta-bar-delta (EDBD) [52], are used to train the MLPs. However, in this paper, thirteen

learning algorithms, conjugate gradient of Fletcher-Reeves (CGFR) [53], Levenberg-Marquardt (LM) [54,55], scaled conjugate gradient (SCG) [56], resilient backpropagation (RP) [57], Broyden-Fletcher-Goldfarb-Shanno (BFGS) [58], conjugate gradient of Powell-Beale (CGPB) [59,60], conjugate gradient of Polak-Ribiére (CGPR) [61], bayesian regularization (BR) [62], one-step secant (OSS) [63], backpropagation with adaptive learning rate (BPALR) [61], backpropagation with momentum (BPM) [61], directed random search (DRS) [64] and genetic algorithm (GA) [65,66] are used to train the MLPs. The radial basis function network is trained by extended delta-bar-delta (EDBD) algorithm. The main aims of this paper are

- to calculate the bandwidth of electrically thin and thick rectangular MSAs by using the MLPs and RBFNs architectures;
- to train the MLPs by the CGFR, LM, SCG, RP, BFGS, CGPB, CGPR, BR, OSS, BPALR, BPM, DRS, and GA, and to train the RBFNs by the EDBD algorithm;
- to compare the bandwidth results of neural models presented in this paper with the results of the conventional methods available in the literature;
- to compare also the bandwidth results of neural models presented in this paper with the results of fuzzy inference systems [36] trained by the improved tabu search algorithm (ITSA) [67], the modified tabu search algorithm (MTSA) [68] and the classical tabu search algorithm (CTSA) [69,70], and with the results of the neural models [35] trained by the BP, DBD, QP, and EDBD algorithms;
- to determine the most appropriate neural model in calculating the bandwidth of rectangular MSAs; and
- to show the superiority of artificial intelligence techniques such as neural networks and fuzzy inference systems over the conventional methods.

In the following sections, the bandwidth of the MSAs, the ANNs, the MLPs and the RBFNs are described briefly, and the application of neural networks to the calculation of the bandwidth of a MSA is then explained.

2. BANDWIDTH OF A RECTANGULAR MICROSTRIP ANTENNA

Figure 1 illustrates a rectangular patch of width W and length L over a ground plane with a substrate of thickness h and a relative dielectric constant ε_r. The bandwidth of this antenna can be written as [1]

$$BW = \frac{s-1}{Q_T \sqrt{s}} \qquad (1)$$
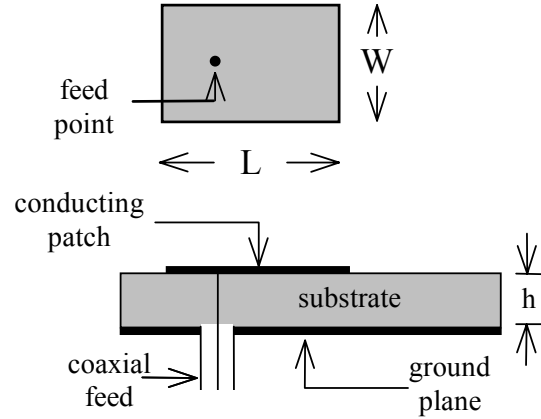
Figure 1. Geometry of rectangular microstrip antenna.

where s is voltage standing wave ratio (VSWR), and Q_T is the total quality factor. The total quality factor, Q_T, can be written as

$$\frac{1}{Q_T} = \left[\frac{1}{Q_r} + \frac{1}{Q_c} + \frac{1}{Q_d} + \frac{1}{Q_s} \right] \qquad (2)$$

where the four terms represent the radiation quality factor, the quality factors due to conductor loss, dielectric loss, and surface wave.

Bandwidth was defined by Pozar [23] as the half-power width of the equivalent circuit impedance response. For a series-type resonance, this bandwidth is

$$BW = \frac{2R}{W_r \dfrac{dX}{dw}\bigg|_{w_r}} \qquad (3)$$

where $Z = R + jX$ is the input impedance at the radian resonant frequency w_r. For a parallel-type resonance, (3) is used with R replaced by G and X replaced by B, where $Y = G + jB$ is the input admittance at resonance. The derivative in (3) can be evaluated by calculating the input impedance at two frequencies near resonance and using a finite difference approximation. The resonant resistance, R, is given by

$$R = R_r + R_d + R_c + R_s \qquad (4)$$

where the four terms represent the radiation resistance, the equivalent resistance of the dielectric loss, the equivalent resistance of the conductor loss, and surface wave radiation resistance. The certain way of calculating the total quality factor and the resonant resistance of both electrically thin and thick

rectangular microstrip patch antennas involves the complicated Green function methods and integral transformation techniques. These methods and techniques suffer from a lack of computational efficiency, which in practice can restrict their usefulness because of high computational time and costs.

In this work, a new technique based on the ANNs for solving this problem efficiently is presented. First, the antenna parameters related to the bandwidth are determined, then the bandwidth depending on these parameters is calculated by using the ANNs.

The feeding method or position is not considered in calculating the bandwidth because the feeding method or position does not effect the intrinsic patch bandwidth. The bandwidth of a patch is significantly greater than that of a printed dipole, at least over the range for which the patch actually resonates ($h<0.12\lambda_0$, where λ_0 is the free space wavelength at the resonant frequency f_r). This fact is consistent with the antenna gain/bandwidth relation to antenna size. Therefore, the effect of the patch width W on the bandwidth of rectangular microstrip antennas must be taken into consideration in the bandwidth calculation of these antennas. From the results of the methods available in the literature [1-36] we see that for a given frequency, larger bandwidth is possible by choosing a thicker substrate and a wider patch. The results also indicate that a lower value of ε_r results in a larger bandwidth.

It is clear from the methods and formulas presented by [1-36] that only three parameters, h/λ_d, W, and the dielectric loss tangent $tan\delta$, are needed to describe the bandwidth. The wavelength in the dielectric substrate, λ_d, is given as

$$\lambda_d = \frac{\lambda_0}{\sqrt{\varepsilon_r}} = \frac{c}{f_r\sqrt{\varepsilon_r}} \tag{5}$$

where c is the velocity of electromagnetic waves in free space.

3. ARTIFICIAL NEURAL NETWORKS (ANNs)

ANNs are biologically inspired computer programs designed to simulate the way in which the human brain processes information. ANNs gather their knowledge by detecting the patterns and relationships in data and learn (or are trained) through experience, not from programming. An ANN is formed from hundreds of single units, artificial neurons or processing elements connected with weights, which constitute the neural structure and are organised in layers. The power of neural computations comes from weight connection in a network. Each neuron has weighted inputs, summation function, transfer

function and one output. The behaviour of a neural network is determined by the transfer functions of its neurons, by the learning rule, and by the architecture itself. The weights are the adjustable parameters and, in that sense, a neural network is a parameterised system. The weighted sum of the inputs constitutes the activation of the neuron. The activation signal is passed through a transfer function to produce the output of a neuron. Transfer function introduces non-linearity to the network. During training, the inter-unit connections are optimised until the error in predictions is minimised and the network reaches the specified level of accuracy. Once the network is trained, new unseen input information is entered to the network to calculate the output for test. ANN represents a promising modelling technique, especially for data sets having non-linear relationships that are frequently encountered in engineering. In terms of model specification, artificial neural networks require no knowledge of the data source but, since they often contain many weights that must be estimated, they require large training sets. In addition, ANNs can combine and incorporate both literature-based and experimental data to solve problems.

There are many types of neural networks for various applications available in the literature [37-40,71]. RBFNs and MLPs are examples of feed-forward networks and both universal approximators. In spite of being different networks in several important respects, these two neural network architectures are capable of accurately mimicking each other [40].

3.1. Multilayered Perceptrons (MLPs)

Multilayered perceptrons (MLPs) [40,49] are the simplest and therefore most commonly used neural network architectures. They have been adapted for the calculation of the bandwidth of the MSA. MLPs can be trained using many different learning algorithms [37-40,71]. In this paper, MLPs are trained with the CGFR, LM, SCG, RP, BFGS, CGPB, CGPR, BR, OSS, BPALR, BPM, DRS, and GA. As shown in Figure 2, an MLP consists of three layers: an input layer, an output layer and an intermediate or hidden layer. Neurons (indicated in Figure 2 with the circle) in the input layer only act as buffers for distributing the input signals x_i to neurons in the hidden layer. Each neuron j in the hidden layer sums up its input signals x_i after weighting them with the strengths of the respective connections w_{ji} from the input layer and computes its output y_j as a function f of the sum, viz.,

$$y_j = f(\sum w_{ji}x_i) \tag{6}$$

f can be a simple threshold function, a sigmoidal or
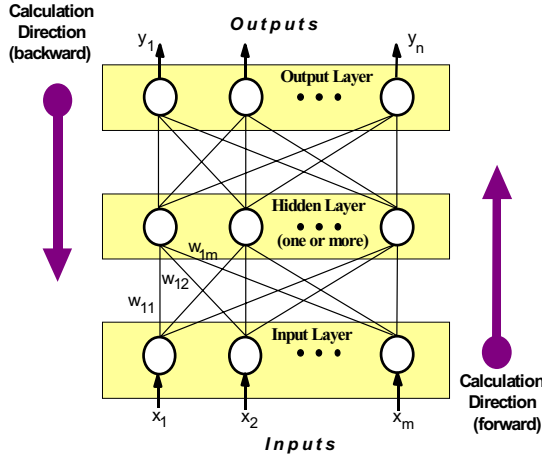
Figure 2. General form of multilayered perceptrons.

hyperbolic tangent function. The output of neurons in the output layer is computed similarly.

Training a network consists of adjusting weights of the network using the different learning algorithms. A learning algorithm gives the change $\Delta w_{ji}(k)$ in the weight of a connection between neurons i and j at time k. The weights are then updated according to the following formula

$$w_{ji}(k+1) = w_{ji}(k) + \Delta w_{ji}(k+1) \qquad (7)$$

3.2. Radial Basis Function Networks (RBFNs)

An alternative network architecture to the MLP is the RBFN [72-74]. A network with an internal representation of hidden neurons, radially symmetric, is named as a RBFN. The topology of the RBFN is obviously similar to that of the three-layered MLP, and the differences lie in the characteristics of the hidden neurons. The structure of a RBFN is shown in Figure 3.

The construction of a RBFN in its most basic form involves three entirely different layers. The input layer is made up of source neurons. The second layer is a hidden layer of high dimension serving a different purpose from that in a MLP. This layer consists of an array of neurons. Each neuron contains a parameter vector called a centre. The neuron calculates the Euclidean distance between the centre and the network input vector, and passes the result through a non-linear function. The output layer is essentially a set of linear combiners and supplies the response of the network. The transformation from input layer to the hidden layer is non-linear, whereas the transformation from the hidden layer to the output layer is linear.
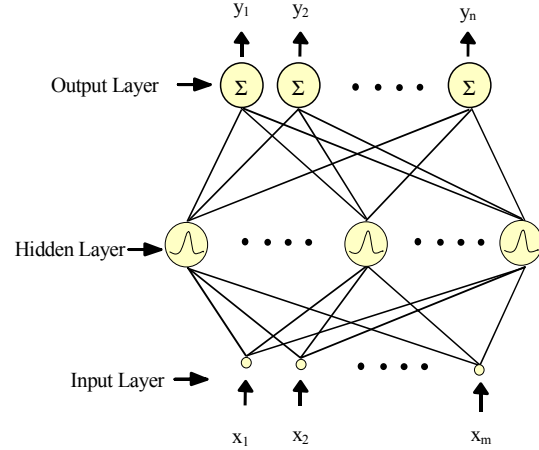
Figure 3. Radial basis function network.

The output of an hidden layer is a function of the distance between the input vector and the stored centre and calculated as

$$O_k = \left\| X - C_k \right\| = \sqrt{\sum_{i=1}^{N} \left(X_i - C_{ki} \right)^2} \qquad (8)$$

The learning consists of using a clustering algorithm for determining the cluster centres (C_k) and a nearest neighbour heuristic for determining the cluster centres. Linear regression, or a gradient descent algorithm is used to determine the weights from the hidden layer to the output layer. In this work, EDBD algorithm is used to train the weights of the layer.

4. NEURAL NETWORKS FOR BANDWIDTH COMPUTATION

ANNs have been adapted for the calculation of the bandwidth (BW) of electrically thin and thick rectangular microstrip antennas. MLPs are trained with the use of CGFR, LM, SCG, RP, BFGS, CGPB, CGPR, BR, OSS, BPALR, BPM, DRS, and GA algorithms. RBFN is trained by using EDBD algorithm. For the neural models, the inputs are h/λ_d, W, and tanδ, and the output is the measured bandwidth BW$_{ME}$. A neural model used in calculating the BW is shown in Figure 4.

For the MLPs trained by DRS and GA, input layer has the linear transfer function, the hidden and output layers have the sigmoid function. For the MLPs trained by the other learning algorithms, the input and output layers have the linear transfer function and the hidden layers have the tangent hyperbolic function. In the RBFNs, the sigmoid function was used for the output layer. Training an ANN with the use of a learning algorithm to compute the bandwidth
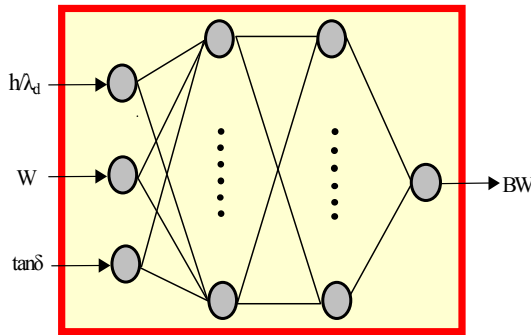
Figure 4. Neural model for bandwidth computation.

involves presenting it sequentially with different sets (h/λ_d, W, $\tan\delta$) and corresponding measured values BW_{ME}. Differences between the target output BW_{ME} and the actual output of the ANNs are evaluated by a learning algorithm. The adaptation is carried out after the presentation of each set (h/λ_d, W, $\tan\delta$) until the calculation accuracy of the network is deemed satisfactory according to some criterion (for example, when the error between BW_{ME} and the actual output for all the training set falls below a given threshold) or the maximum allowable number of epochs or generations is reached.

The training and test data sets used in this paper have been obtained from the previous experimental works [33,34], and are given in Table 1. The 27 data sets in Table 1 were used to train the networks. 6 test data sets which are marked with an asterisk in Table 1 were used for test. The number of neurons in the hidden layers and train epochs for neural models presented here are given in Table 2. 10x7x8 in Table 2 means that the number of neurons was 10, 7, and 8 for the first, second, and third hidden layers, respectively. Initial weights of the neural models were set up randomly.

5. RESULTS AND CONCLUSIONS

The bandwidths calculated by using neural models presented in this paper for electrically thin and thick rectangular microstrip patch antennas are listed in Table 3. For comparison, the results obtained by using the conventional methods [1,21,31-33], and the neural models presented by [35] and the fuzzy inference systems [36] are given in Table 4. EDBD, DBD, BP, QP, ITSA, CTSA, and MTSA in Table 4 represent, respectively, the bandwidths calculated by the neural models [35] trained by EDBD, DBD, BP, QP, and calculated by the fuzzy inference systems [36] trained by ITSA, CTSA, and MTSA. The total absolute errors between the computed and experimental results for neural models, fuzzy inference systems, and conventional methods are listed in Table 5 and Table 6.

Table 1. Measured bandwidth results and dimensions for electrically thin and thick rectangular microstrip antennas.

Patch No	h (mm)	f_r (GHz)	h/λ_d	W (mm)	$\tan\delta$	Measured [33,34] BW_{ME} (%)
1	0.17	7.740	0.0065	8.50	0.001	1.07
2	0.79	3.970	0.0155	20.00	0.001	2.20
3	0.79	7.730	0.0326	10.63	0.001	3.85
4	0.79	3.545	0.0149	20.74	0.002	1.95
5	1.27	4.600	0.0622	9.10	0.001	2.05
6	1.57	5.060	0.0404	17.20	0.001	5.10
7	1.57	4.805	0.0384	18.10	0.001	4.90*
8	1.63	6.560	0.0569	12.70	0.002	6.80
9	1.63	5.600	0.0486	15.00	0.002	5.70
10	2.00	6.200	0.0660	13.37	0.002	7.70*
11	2.42	7.050	0.0908	11.20	0.002	10.90
12	2.52	5.800	0.0778	14.03	0.002	9.30
13	3.00	5.270	0.0833	15.30	0.002	10.00
14	3.00	7.990	0.1263	9.05	0.002	16.00*
15	3.00	6.570	0.1039	11.70	0.002	13.60
16	4.76	5.100	0.1292	13.75	0.002	15.90
17	3.30	8.000	0.1405	7.76	0.002	17.50
18	4.00	7.134	0.1519	7.90	0.002	18.20*
19	4.50	6.070	0.1454	9.87	0.002	17.90
20	4.76	5.820	0.1475	10.00	0.002	18.00
21	4.76	6.380	0.1617	8.14	0.002	19.00
22	5.50	5.990	0.1754	7.90	0.002	20.00
23	6.26	4.660	0.1553	12.00	0.002	18.70
24	8.45	4.600	0.2091	7.83	0.002	20.90
25	9.52	3.580	0.1814	12.56	0.002	20.00
26	9.52	3.980	0.2017	9.74	0.002	20.60
27	9.52	3.900	0.1976	10.20	0.002	20.30*
28	10.00	3.980	0.2119	8.83	0.002	20.90
29	11.00	3.900	0.2284	7.77	0.002	21.96
30	12.00	3.470	0.2216	9.20	0.002	21.50
31	12.81	3.200	0.2182	10.30	0.002	21.60
32	12.81	2.980	0.2032	12.65	0.002	20.40
33	12.81	3.150	0.2148	10.80	0.002	21.20*

*Test data sets

When the performances of neural models presented in this paper and in [35] are compared with each other, the best results for training and test were obtained from the MLP network trained by the CGPB and BFGS, respectively, as shown in Table 5. However, among the neural models, the highest accuracy in the total absolute errors was achieved with the CGFR algorithm. When the two heuristic approaches were compared with each other, the results of DRS were found better than those of the GA. It is also clear from Table 5 that in most cases the results of neural models presented in this paper are better than those of the neural models presented by [35] and that the best result in the total absolute errors is obtained from the fuzzy inference systems trained by ITSA. However, the train absolute error of the fuzzy inference systems trained by ITSA is larger than that of the MLPs trained by CGFR, LM, SCG, CGPB, and CGPR algorithms.

Table 2. The ANN configurations and the number of train epochs for neural models presented in this paper.

ANN Architectures/ Algorithms		The number of neurons in the hidden layers	The number of train epochs
MLPs	CGFR	10 x 7 x 8	2 500
	LM	6 x 3	201
	SCG	11 x 8 x 7	1 200
	RP	12 x 10	6 500
	BFGS	10 x 5	700
	CGPB	7 x 7 x 4	1 500
	CGPR	7 x 7 x 4	1 500
	BR	3 x 4 x 3	290
	OSS	10 x 8 x 8	2 500
	BPALR	45 x 35 x 35	2 500
	BPM	45 x 35 x 35	5 000
	DRS	12 x 6	740
	GA	20 x 25	1 850
RBFN	EDBD	20 x 6	185 200

It can be clearly seen from Tables 4 and 6 that the conventional methods give comparable results-some cases are in very good agreement with measurements, and others are far off. When the results of neural models and fuzzy inference systems are compared with the results of the conventional methods, the results of all neural models and fuzzy inference systems are better than those predicted by the conventional methods. The very good agreement between the measured bandwidth values and the computed bandwidth values of neural models and fuzzy inference systems supports the validity of the artificial intelligence techniques and also illustrates the superiority of artificial intelligence techniques over the conventional methods.

A distinct advantage of neural computation is that, after proper training, a neural network completely bypasses the repeated use of complex iterative processes for new cases presented to it. For engineering applications, the simple models are very usable. Thus the neural models given in this work can also be used for many engineering applications and purposes.

REFERENCES

1. I. J. Bahl and P. Bhartia, *Microstrip Antennas*, Artech House, Dedham, MA, 1980.
2. J. R. James, P. S. Hall and C. Wood, *Microstrip Antennas-Theory and Design*, Peter Peregrisnus Ltd., London, 1981.
3. G. Dubost, *Flat Radiating Dipoles and Applications to Arrays*, Research Studies Press, 1981.
4. J. R. Mosig and F. E. Gardiol, "A Dynamic Radiation Model for Microstrip Structures", in *Advances in Electronics and Electron Physics*, Academic Press, New York, Vol. 59, pp.139-227, 1982.
5. R. E. Munson, "Microstrip Antennas," in *Antenna Engineering Handbook*, R.C. Johnson (Editor), Mc Graw-Hill, New York, 1983.
6. D. M. Pozar, *Antenna Design Using Personal Computers*, Artech House, Dedham, MA, pp.121-126, 1985.
7. K. C. Gupta and A. Benalla (Editors), *Microstrip Antenna Design*, Artech House, MA, 1988.
8. W. F. Richards, "Microstrip Antennas," in *Antenna Handbook*, Y.T. Lo and S.W. Lee (Editors), Van Nostrand Reinhold, New York, 1988.
9. J. R. James and P. S. Hall, *Handbook of Microstrip Antennas*, IEE Electromagnetic Wave Series, Peter Peregrinus Ltd., London, Vols. 1 and 2, No. 28, 1989.
10. Y. T. Lo, S. M. Wright and M. Davidovitz, "Microstrip Antennas," in *Handbook of Microwave and Optical Components*, K. Chang (Editor), John Wiley and Sons, New York, Vol. 1, pp.764-889, 1989.
11. K. F. Lee and J. S. Dahele, "Characteristics of Microstrip Patch Antenna and Some Methods of Improving Frequency Agility and Bandwitdh," in *Handbook of Microstrip Antennas*, J.R. James and P.S. Hall (Editors), IEE Electromagnetic Wave Series, Peter Peregrinus Ltd., London, Chapter 3, Vol. 1, No. 28, 1989.
12. P. Bhartia, K. V. S. Rao and R. S. Tomar (Editors), *Millimeter-Wave Microstrip and Printed Circuit Antennas*, Artech House, Canton, MA, 1991.
13. K. Hirasawa and M. Haneishi, *Analysis, Design, and Measurement of Small and Low-Profile Antennas*, Artech House, Canton, MA, 1992.
14. D. M. Pozar and D. H. Schaubert (Editors), *Microstrip Antennas-The Analysis and Design of Microstrip Antennas and Arrays*, IEEE Press, New York, 1995.
15. J. F. Zurcher and F. E. Gardiol, *Broadband Patch Antennas*, Artech House, Norwood, MA, 1995.
16. R. A. Sainati, *CAD of Microstrip Antennas for Wireless Applications*, Artech House, Boston, 1996.
17. K. F. Lee and W. Chen, *Advances in Microstrip and Printed Antennas*, John Wiley and Sons, 1997.
18. R. Garg, P. Bhartia, I. J. Bahl and A. Ittipiboon, *Microstrip Antenna Design Handbook*, Artech House, Boston, 2001.
19. J. Vandensande, H. Pues and A. Van De Capelle, "Calculation of the Bandwidth of Microstrip Resonator Antennas," in *Proc. of 9th European Conference*, Brighton, England, pp.116-119, Sept. 1979.

Table 3. Comparison of measured and calculated bandwidths obtained by using neural models presented in this paper for electrically thin and thick rectangular microstrip antennas.

Patch No	Measured BW_{ME} (%) [33,34]	Present Neural Models													
		MLP													RBFN
		CGFR	LM	SCG	RP	BFGS	CGPB	CGPR	BR	OSS	BPALR	BPM	DRS	GA	EDBD
1	1.070	1.069	1.071	1.071	1.070	1.070	1.070	1.070	1.070	1.067	1.071	1.068	1.400	1.573	1.048
2	2.200	2.199	2.200	2.200	2.201	2.202	2.200	2.200	2.200	2.203	2.200	2.201	2.182	2.620	2.292
3	3.850	3.850	3.850	3.850	3.850	3.850	3.851	3.850	3.850	3.853	3.837	3.840	3.336	3.288	3.849
4	1.950	1.949	1.950	1.949	1.950	1.952	1.950	1.950	1.950	1.948	1.945	1.949	1.951	1.943	1.899
5	2.050	2.050	2.050	2.050	2.051	2.048	2.050	2.050	2.050	2.049	2.061	2.062	2.210	2.120	2.077
6	5.100	5.101	5.100	5.100	5.099	5.100	5.099	5.100	5.100	5.100	5.097	5.100	5.223	4.816	5.024
7	4.900*	4.560	4.900	4.766	4.922	4.764	4.016	4.011	5.175	4.137	4.233	4.300	4.571	4.506	4.437
8	6.800	6.800	6.800	6.800	6.800	6.798	6.801	6.800	6.799	6.811	6.790	6.801	6.754	7.076	6.744
9	5.700	5.699	5.700	5.700	5.700	5.700	5.702	5.700	5.701	5.702	5.694	5.701	5.632	5.470	5.806
10	7.700*	7.811	7.763	7.862	8.132	7.639	7.719	7.691	7.869	7.760	7.705	7.536	7.891	8.061	7.968
11	10.900	10.899	10.901	10.903	10.900	10.926	10.900	10.900	10.900	10.910	10.902	10.906	11.285	11.250	11.080
12	9.300	9.299	9.299	9.305	9.299	9.301	9.300	9.301	9.302	9.271	9.301	9.298	9.425	9.451	9.471
13	10.000	10.001	10.001	9.999	10.001	9.999	9.999	9.999	9.998	10.016	9.980	9.997	9.983	9.864	9.813
14	16.000*	15.954	15.918	16.161	15.995	15.890	16.063	16.100	16.337	16.396	15.982	16.182	15.924	16.167	15.940
15	13.600	13.601	13.599	13.595	13.605	13.548	13.600	13.600	13.600	13.598	13.576	13.593	13.169	13.135	13.225
16	15.900	15.899	15.901	15.899	15.902	15.919	15.902	15.901	15.905	15.900	15.905	15.906	16.003	16.086	16.106
17	17.500	17.499	17.504	17.496	17.493	17.496	17.499	17.502	17.501	17.482	17.494	17.499	17.284	17.516	17.417
18	18.200*	18.345	18.422	18.217	18.179	18.365	18.297	18.300	18.311	18.395	18.458	18.537	18.340	18.394	18.381
19	17.900	17.877	17.847	17.853	17.824	17.850	17.871	17.869	17.860	17.864	17.861	17.890	17.947	17.883	17.996
20	18.000	18.023	18.055	18.048	18.066	18.051	18.035	18.034	18.019	18.049	18.036	18.010	18.129	18.046	18.170
21	19.000	19.004	19.004	19.013	19.026	19.016	18.999	18.999	19.037	19.055	19.098	19.146	19.103	19.050	19.035
22	20.000	20.000	19.991	19.998	19.991	19.989	20.000	20.000	19.974	19.919	19.801	19.764	19.893	19.801	19.733
23	18.700	18.699	18.696	18.695	18.696	18.797	18.698	18.697	18.699	18.690	18.709	18.719	18.572	18.419	18.655
24	20.900	20.919	20.897	20.897	20.893	20.887	20.903	20.906	20.888	20.927	21.009	21.016	21.135	21.136	21.076
25	20.000	20.000	20.000	20.005	20.000	19.819	20.002	20.000	19.989	19.999	19.902	19.902	19.859	19.807	19.792
26	20.600	20.600	20.592	20.606	20.579	20.610	20.596	20.604	20.667	20.612	20.596	20.593	20.969	20.917	20.797
27	20.300*	20.237	20.455	20.562	20.429	20.484	20.505	20.517	20.604	20.512	20.397	20.382	20.818	20.796	20.638
28	20.900	20.873	20.910	20.889	20.921	20.913	20.891	20.883	20.841	20.958	21.046	21.033	21.247	21.190	21.175
29	21.960	21.948	21.953	21.963	21.934	21.941	21.952	21.955	21.941	21.939	21.818	21.861	21.521	21.702	21.782
30	21.500	21.545	21.520	21.506	21.573	21.583	21.522	21.523	21.571	21.503	21.552	21.526	21.439	21.435	21.513
31	21.600	21.571	21.590	21.596	21.541	21.525	21.591	21.589	21.566	21.557	21.422	21.444	21.298	21.385	21.339
32	20.400	20.405	20.401	20.400	20.414	20.494	20.401	20.401	20.402	20.403	20.528	20.520	20.499	20.646	20.479
33	21.200*	21.265	21.493	21.481	21.370	21.270	21.431	21.431	21.279	21.407	21.197	21.218	21.166	21.309	21.180

20. A. G. Derneryd and A. G. Lind, "Extended Analysis of Rectangular Microstrip Resonator Antenna," *IEEE Trans. Ant. Propagat.*, pp.846-849, 1979.

21. K.R. Carver and J.W. Mink, "Microstrip Antenna Technology", *IEEE Trans. Ant. Propagat.*, Vol. AP-29, pp. 2-24, 1981.

22. W. F. Richards, Y. T. Lo and D. D. Harrison, "An Improved Theory for Microstrip Antennas and Applications," *IEEE Trans. Ant. Propagat.*, Vol. AP-29, pp. 38-46, Jan. 1981.

23. D. M. Pozar, "Considerations for Millimeter Wave Printed Antennas," *IEEE Trans. Ant. Propagat.*, Vol. AP-31, pp.740-747, Sept. 1983.

24. G. Kumar and K. C. Gupta, "Broadband Microstrip Antennas Using Additional Resonators Gap Coupled to the Radiating Edges," *IEEE Trans. Ant. Propagat.*, Vol. AP-32, pp.1375-1379, Dec. 1984.

25. H. F. Pues and A. R. Van De Capelle, "Accurate Transmission-Line Model fort the Rectangular Microstrip Antennas," *IEE Proc. Pt. H.*, Vol. 13, No. 6, pp.334-340, 1984.

26. P. Perlmutter, S. Shtrikman, and D. Treves, "Electric Surface Current Model for the Analysis of Microstrip Antennas with Application to Rectangular Elements," *IEEE Trans. Ant. Propagat.*, Vol. AP-33, pp.301-311, March 1985.

27. K. Bhattacharyya and R. Garg, "Effect of Substrate on the Efficiency of an Arbitrarily Shaped Microstrip Patch Antenna," *IEEE Trans. Ant. Propagat.*, Vol. AP-34, pp.1181-1188, 1986.

28. E. Chang, S.A. Long and W. F. Richards, "An Experimental Investigation of Electrically Thick Rectangular Microstrip Antennas," *IEEE Trans. Ant. Propagat.*, Vol. AP-34, pp.767-772, 1986.

29. D. M. Pozar and S. M. Voda, "A Rigorous Analysis of a Microstrip Line Fed Patch Antenna," *IEEE Trans. Ant. Propagat.*, Vol. AP-35, pp.1343-1350, 1987.

30. H. F. Pues and A. R. Van De Capelle, "An Impedance Matching Technique for Increasing the Bandwidth of Microstrip Antennas," *IEEE Trans. Ant. Propagat.*, Vol. AP-37, pp.1345-1354, Nov. 1989.

Table 4. Bandwidths obtained from the conventional methods and artificial intelligence techniques available in the literature for electrically thin and thick rectangular microstrip antennas.

Patch No	Measured BW$_{ME}$ (%) [33,34]	Conventional Methods in the Literature					Artificial Intelligence Techniques in the Literature						
							Neural Models [35]				Fuzzy Inference Systems [36]		
		[21]	[1]	[31]	[33]	[32]	EDBD	DBD	BP	QP	ITSA	CTSA	MTSA
1	1.070	0.82	0.84	0.30	1.20	0.26	1.081	1.068	1.178	1.271	1.070	1.070	1.070
2	2.200	1.45	2.03	0.87	2.78	0.75	2.193	2.197	2.304	2.117	2.200	2.200	2.200
3	3.850	2.99	3.76	1.88	5.03	1.64	3.840	3.854	3.670	3.753	3.848	3.850	3.850
4	1.950	1.29	1.69	0.72	2.46	0.61	1.948	1.948	1.905	2.034	1.950	1.959	1.950
5	2.050	1.54	1.90	0.72	4.09	0.84	2.046	2.047	2.117	2.612	2.051	2.050	2.050
6	5.100	4.21	5.14	2.67	6.46	2.35	4.945	5.340	5.211	4.837	5.101	5.100	5.100
7	4.900	3.96	4.87	2.51	6.17	2.20	4.916	4.898	4.831	4.854	4.899	4.900	4.895
8	6.800	5.98	6.70	3.69	8.12	3.43	6.824	6.788	6.887	6.757	6.775	6.595	6.798
9	5.700	4.76	5.69	3.02	7.12	2.78	5.679	5.718	5.822	5.783	5.699	5.676	5.711
10	7.700	7.29	7.81	4.41	9.16	4.20	8.006	7.865	7.727	7.730	7.759	7.877	7.769
11	10.900	11.31	10.88	6.39	11.72	6.50	10.858	10.901	11.040	10.998	10.906	11.217	10.896
12	9.300	9.14	9.26	5.36	10.42	5.26	9.336	9.287	9.155	9.085	9.255	9.476	9.287
13	10.000	10.30	10.14	5.88	11.15	5.83	9.990	10.000	10.092	10.131	10.003	9.860	9.994
14	16.000	18.42	15.64	9.41	15.16	10.36	15.975	15.862	15.940	15.851	16.005	15.998	16.139
15	13.600	13.84	12.75	7.53	13.14	7.90	13.607	13.601	13.528	13.388	13.598	13.174	13.600
16	15.900	18.06	15.73	9.35	15.11	10.50	15.881	15.917	15.994	16.100	15.914	16.050	15.905
17	17.500	15.29	18.48	8.39	17.00	11.28	17.523	17.480	17.349	17.264	17.450	17.442	17.324
18	18.200	13.62	20.09	8.15	17.77	12.18	18.254	18.433	18.372	18.339	18.288	18.357	18.284
19	17.900	14.54	19.17	8.31	17.34	11.70	17.844	17.917	17.949	17.947	17.845	17.884	17.797
20	18.000	14.08	19.46	8.19	17.47	11.80	18.016	18.091	18.101	18.129	18.060	18.050	17.977
21	19.000	12.45	21.47	7.95	18.42	12.93	19.113	19.054	19.113	19.094	18.955	18.988	19.110
22	20.000	10.73	23.41	7.63	19.29	14.10	19.818	19.766	19.878	19.883	19.999	19.714	19.955
23	18.700	13.01	20.55	8.10	18.01	12.57	18.804	18.620	18.433	18.599	18.690	18.603	18.688
24	20.900	7.85	28.24	6.76	21.26	16.49	21.009	21.101	21.170	21.163	20.896	21.080	20.917
25	20.000	10.10	24.27	7.46	19.66	14.54	19.851	19.842	19.857	19.836	19.997	19.790	20.035
26	20.600	8.45	27.17	7.02	20.85	16.10	20.608	20.760	20.916	20.900	20.602	20.759	20.478
27	20.300	8.76	26.59	7.10	20.61	15.76	20.524	20.608	20.724	20.734	20.296	20.599	20.274
28	20.900	7.63	28.64	6.67	21.40	16.65	20.977	21.147	21.241	21.231	20.909	21.145	21.056
29	21.960	6.50	31.03	6.14	22.26	17.56	21.885	21.777	21.557	21.609	21.960	21.741	21.973
30	21.500	6.92	30.06	6.32	21.91	17.13	21.495	21.469	21.412	21.433	21.510	21.461	21.580
31	21.600	7.11	29.56	6.41	21.73	16.95	21.535	21.317	21.342	21.249	21.566	21.309	21.377
32	20.400	8.26	27.39	6.90	20.92	16.07	20.500	20.592	20.569	20.498	20.401	20.526	20.514
33	21.200	7.39	29.07	6.54	21.55	16.77	21.460	21.184	21.148	21.103	21.221	21.178	21.173

Table 5. Train, test and total absolute errors between the measured and calculated bandwidhs for various neural networks and fuzzy inference systems.

Artificial Intelligence Techniques		Learning Algorithms	Train Absolute Errors (%)	Test Absolute Errors (%)	Total Absolute Errors (%)
Present Neural Models	MLP	CGFR	0.199	0.770	0.969
		LM	0.194	0.815	1.009
		SCG	0.174	1.017	1.191
		RP	0.421	0.779	1.200
		BFGS	0.824	0.726	1.550
		CGPB	0.136	1.499	1.635
		CGPR	0.141	1.546	1.687
		BR	0.410	1.275	1.685
		OSS	0.499	1.833	2.332
		BPALR	1.345	1.048	2.393
		BPM	1.229	1.383	2.612
		DRS	5.044	1.288	6.332
		GA	6.069	1.721	7.790
	RBFN	EDBD	3.633	1.330	4.963
Fuzzy Interference Systems [36]		ITSA	0.384	0.178	0.562
		MTSA	1.270	0.350	1.620
		CTSA	3.435	0.657	4.092
Neural Models in the Literature [35]	MLP	EDBD	1.430	0.885	2.315
		DBD	2.267	0.862	3.129
		BP	4.158	0.804	4.962
		QP	4.921	0.895	5.816

Table 6. The total absolute errors between the measured and calculated bandwidths for the conventional methods in the literature.

Conventional Methods in the Literature	[21]	[1]	[31]	[33]	[32]
Total absolute deviations from the measured data (%)	178.69	88.76	266.93	23.92	140.02

31. D. R. Jackson and N. G. Alexopoulos, "Simple Approximate Formulas for Input Resistance, Bandwidth, and Efficiency of a Resonant Rectangular Patch," *IEEE Trans. Ant. Propagat.*, Vol. 39, pp.407-410, 1991.

32. K. Guney, "Bandwidth of a Resonant Rectangular Microstrip Antenna," *Microwave and Optical Technology Letters*, Vol. 7, pp.521-524, 1994.

33. M. Kara, "A Novel Technique to Calculate the Bandwidth of Rectangular Microstrip Antenna Elements with Thick Substrates," *Microwave and Optical Technology Letters*, Vol. 12, pp.59-64, 1996.

34. M. Kara, "A Simple Technique for the Calculation of Bandwidth of Rectangular Microstrip Antenna Elements with Various Substrate Thicknesses," *Microwave and Optical Technology Letters*, Vol. 12, pp.16-20, 1996.

35. S. Sagiroglu, K. Guney and M. Erler, "Calculation of Bandwidth for Electrically Thin and Thick Rectangular Microstrip Antennas With The Use of Multilayered Perceptrons," *International Journal of RF and Microwave Computer-Aided Engineering*, Vol. 9, pp.277-286, 1999.

36. A. Kaplan, K. Guney and S. Özer, "Fuzzy Associative Memories for the Computation of the Bandwidth of Rectangular Microstrip Antennas with Thin and Thick Substrates," *International Journal of Electronics (IJE)*, Vol. 88, pp.189-195, 2001.

37. Q. J. Zhang and K. C. Gupta, *Neural Networks for RF and Microwave Design*, Artech House, Boston, MA, 2000.

38. C. G. Christodoulou and M. Georgiopoulos, *Application of Neural Networks in Electromagnetics*, Artech House, MA, 2001.

39. A. C. Maren, C. Harston and R. Pap, *Handbook of Neural Computing Applications*, Academic Press, London, 1990.

40. S. Haykin, *Neural Networks: A Comprehensive Foundation*, Macmillan College Publishing Company, New York, 1994.

41. S. Sagiroglu and K. Guney, "Calculation of Resonant Frequency for an Equilateral Triangular Microstrip Antenna Using Artificial Neural Networks," *Microwave and Optical Technology Letters*, Vol.14, No.2, pp.89-93, 1997.

42. D. Karaboga, K. Guney, S. Sagiroglu and M. Erler, "Neural Computation of Resonant Frequency of Electrically Thin and Thick Rectangular Microstrip Antennas," *IEE Proc. Microw. Antennas Propag.*, Vol. 146, No. 2, pp.155-159, April 1999.

43. S. Sagiroglu, K. Guney and M. Erler, "Neural Computation of Mutual Coupling Coefficient of Electrically Thin and Thick Rectangular Microstrip Antennas," *Proc. of International Conference on Neural Network and Brain (NN&B'98)*, Beijing, China, pp.223-226, Oct. 1998.

44. S. Sagiroglu, K. Guney and M. Erler, "Resonant Frequency Calculation for Circular Microstrip Antennas Using Artificial Neural Networks," *International Journal of RF and Microwave Computer-Aided Engineering*, Vol. 8, pp.270-277, 1998.

45. K. Guney, M. Erler and S. Sagiroglu, "Neural Computation of Mutual Coupling Coefficient Between Two Rectangular Microstrip Antennas With Various Substrate Thicknesses," *Proc. of PIERS'98*, Nantes, France, p. 57, July 13-17, 1998.

46. K. Guney, M. Erler and S. Sagiroglu, "Artificial Neural Networks for the Resonant Resistance Calculation of Electrically Thin and Thick Rectangular Microstrip Antennas," *Electromagnetics*, Vol. 20, pp.387-400, 2000.

47. K. Guney, S. Sagiroglu and M. Erler, "Comparison of Neural Networks for Resonant Frequency Computation of Electrically Thin and Thick Rectangular Microstrip Antennas," *Journal of Electromagnetic Waves and Applications (JEWA)*, Vol. 15, pp.1121-1145, 2001.

48. K. Guney, S. Sagiroglu and M. Erler, "Generalized Neural Method to Determine Resonant Frequencies of Various Microstrip Antennas," *International Journal of RF and Microwave Computer-Aided Engineering*, Vol. 12, pp.131-139, 2002.

49. D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing*. Vol.1, The MIT Press, Cambridge, 1986.

50. R. A. Jacobs, "Increased Rate of Convergence Through Learning Rate Adaptation," *Neural Networks*, Vol. 1, pp.295-307, 1988.

51. S. E. Fahlman, "Fast Learning Variations on Back Propagation: An Empirical Study," In D.S. Touretzky, G.E. Hinton and T.J. Sejnowski (Editors), *Proc. of the 1988 Connectionist Models Summer School*, San Mateo, CA, Morgan Kaufmann, pp. 38-51, 1988.

52. A. A. Minai and R. D. Williams, "Acceleration of Backpropagation Through Learning Rate and

Momentum Adaptation," *Int. Joint Conf. on Neural Networks*, Vol. 1, pp.676-679, 1990.

53. R. Fletcher and C. M. Reeves, "Function Minimization by Conjugate Gradients," *Computer Journal*, Vol. 7, pp.149-154, 1964.

54. K. Levenberg, "A Method for the Solution of Certain Nonlinear Problems in Least Squares," *Quart. Appl. Math.*, Vol. 2, pp.164-168, 1944.

55. D. W. Marquardt, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *J. Soc. Ind. Appl. Math.*, Vol. 11, pp.431-441, 1963.

56. M. F. Moller, "A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning," *Neural Networks*, Vol. 6, pp.525-533, 1993.

57. M. Riedmiller and H. Braun, "A Direct Adaptive Method for Faster Backpropagation Learning: the RPROP Algorithm," *Proc. of the IEEE International Conference on Neural Networks*, San Francisco, CA, March, Vol. 1, pp.586-591, 1993.

58. J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Englewood Cliffs, NJ: Prentice-Hall, 1983.

59. M. J. D. Powell, "Restart Procedures for the Conjugate Gradient Method," *Mathematical Programming*, Vol. 12, pp.241-254, 1977.

60. E. M. L. Beale, *A Derivation of Conjugate Gradients, Numerical Methods for Nonlinear Optimization*, London, Academic Press, 1972.

61. M. T. Hagan, H. B. Demuth and M. Beale, *Neural Network Design*, Boston PWS Publishing Company, 1996.

62. D. J. C. MacKay, "Bayesian Interpolation," *Neural Computation*, Vol. 4, pp.415-447, 1992.

63. R. Battiti, "First and Second Order Methods for Learning: Between Steepest Descent and Newton's Method," *Neural Computation*, Vol. 4, pp.141-166, 1992.

64. J. Matyas, "Random Optimization," *Automation and Remote Control*, Vol. 26, pp.246-253, 1965.

65. J. H. Holland, *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, MI, 1975.

66. L. Davis, *Handbook of Genetic Algorithms*, (Eds.), Van Nostrand Reinhold, NY, 1991.

67. S. Ozer, K. Guney and A. Kaplan, "Computation of the Resonant Frequency of Electrically Thin and Thick Rectangular Microstrip Antennas with the Use of Fuzzy Inference Systems," *International Journal of RF and Microwave Computer-Aided Engineering*, Vol. 10, pp.108-119, 2000.

68. D. Karaboga, K. Guney, A. Kaplan and A. Akdaglı, "A New Effective Side Length Expression Obtained Using a Modified Tabu Search Algorithm for the Resonant Frequency of a Triangular Microstrip Antenna," *International Journal of RF and Microwave Computer-Aided Engineering*, Vol. 8, pp.4-10, 1998.

69. F. Glover, "Tabu Search Part I," *ORSA Journal on Computing*, Vol. 1, pp.190-206, 1989.

70. F. Glover, "Tabu Search Part II," *ORSA Journal on Computing*, Vol. 2, pp.4-32, 1990.

71. Neural Computing, *A Technology Handbook for Professional II/PLUS and NeuralWorks Explorer*, NeuralWare, Inc., Technical Publications Group, Pittsburgh, 1996.

72. D. S. Broomhead and D. Lowe, "Multivariable Functional Interpolation and Adaptive Networks," *Complex Systems*, Vol. 2, pp.321-355, 1988.

73. J. Moody and C. Darken, "Fast-Learning in Networks of Locally-Tuned Processing Units," *Neural Computation*, Vol. 1, pp.281-294, 1989.

74. S. Chen, C. F. N. Cowan and P. M. Grant, "Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks," *IEEE Trans. on Neural Networks*, Vol. 4, pp.302-309, 1991.

Sinan Gültekin was born in Konya, Turkey, on September 16, 1963. He received the BS, MS and PhD degrees from Erciyes University, Kayseri, in 1988, 1992 and 2002, all in electronic engineering. Currently, he is a lecturer at the Department of Electric and Electronic Engineering, Selçuk University, Konya, Turkey. His current research activities include neural networks and their applications to antennas.

Kerim Güney was born in Isparta, Turkey, on February 28, 1962. He received the BS degree from Erciyes University, Kayseri, in 1983, the MS degree from Istanbul Technical University, in 1988, and the PhD degree from Erciyes University, in 1991, all in electronic engineering. From 1991 to 1995 he was an assistant professor and now is a professor at the Department of Electronic

Engineering, Erciyes University, where he is working in the areas of optimization techniques (genetic, tabu search, ant colony and simulated annealing algorithms), fuzzy inference systems, neural networks, their applications to antennas, microstrip and horn antennas, and antenna pattern synthesis.

Seref Sagiroglu was born on August 8, 1965, in Antalya, Turkey. He received the BS degree in electronic engineering from Erciyes University, Kayseri, Turkey, in 1987. He received the PhD degree in system engineering from the University of Wales College of Cardiff, UK, in 1994. He is now an associate professor at the Department of Computer Engineering, Erciyes University. His research interests include modern heuristic optimization techniques (genetic algorithms, tabu searches, and simulated annealing), artificial neural networks, fuzzy logic, intelligent system identification, modelling and control, and robotics.

APPLICATION OF NEURAL NETWORKS IN THE ESTIMATION OF TWO-DIMENSIONAL TARGET ORIENTATION

A. Kabiri

Physics Department, Tehran University, Tehran, Iran

N. Sarshar

Electrical Engineering Department, McMaster University, Ontario, Canada

K. Barkeshli

Electrical Engineering Department, Sharif University of Technology, Tehran, Iran
Email: barkeshli@sharif.edu

Abstract

A new method for the robust estimation of target orientation using measured radar cross section is proposed. The method is based on a Generalized Regression Neural Network (GRNN) scheme. The network is trained by the FFT modulus of bistatic radar cross section data sampled at the receiver positions. The target value to be trained is the angle between a defined target orientation and the incident wave. Results based on actual measurements are presented.

INTRODUCTION

Accurate estimation of target orientation is essential in range profiling schemes [1-4]. In such cases, the knowledge of target orientation can yield information about the target-structure. The range profile itself, however, is quite sensitive to variations in target orientation and cannot be the basis for such estimation. A detailed tracking of object orientation is therefore necessary.

Attempts have been made to use artificial neural networks (ANNs) for solving the inverse problem. However, the proposed methods have not been able to exploit the fundamental advantages of neural systems, which are their speed and robustness. In many instances, the problem formulation was fitted into previously developed algorithms for network training [5, 6]. Nevertheless, successful methods were developed for cases where a priori knowledge of the target geometry is available [7]. Neural networks have proven to do well in target classification area. A spectral approach to radar target classification using ANNs was proposed in [8].

The Generalized Regression Neural Network (GRNN) [9] is among radial basis networks and has found applications in regression and function estimation processes. It has been shown that given a sufficient number of neurons in the hidden layer, a GRNN can approximate a continuous function to an arbitrary precision [10].

In this paper, the orientation of a cylindrical conducting target is estimated with a GRNN network using radar cross section data. The definition of the problem is shown in Figure 1, where a target is illuminated by a number of transmitters/receivers at different angles of incidence. The orientation angle is defined as the angle between a preferred direction specified on the target geometry and the incident wave. The task is to find the orientation angle by using a number of bistatic radar measurements.

THE FORWARD PROBLEM

Consider a perfectly conducting cylinder of arbitrary cross section shape, as shown in Figure 2, illuminated by a plane wave in free space. The cylindrical contour is denoted by C. For the TM^z polarization, the electric field integral equation (EFIE) is given by

$$E_z^i(\boldsymbol{\rho}) = \frac{k_0 Z_0}{4} \oint_C K_z(\boldsymbol{\rho}') H_0^{(2)}(k_0 |\boldsymbol{\rho} - \boldsymbol{\rho}'|) d\ell' \quad (1)$$

where $\boldsymbol{\rho}$ and $\boldsymbol{\rho}'$ are the field and source points, respectively, and $H_0^{(2)}$ is the zeroth order Hankel function of the second kind.

The above integral equations are solved numerically by the method of moments. Once the induced current is calculated, the scattering echo width is given by

$$\sigma^{TM} = \frac{k_0 Z_0}{4} | \oint_C K_z(\boldsymbol{\rho}') e^{jk_0(x'\cos\phi + y'\sin\phi)} d\ell' |^2 \quad (2)$$

THE GRNN

The Generalized Regression Neural Network belongs to the family of radial basis neural networks. Radial basis networks require more neurons than standard feed-forward backpropagation networks, but they can often be designed in a fraction of the time it takes to train standard feed-forward networks. They work best when many training vectors are available.

Radial basis networks were previously used in field estimation processes. It is shown that given a sufficient number of neurons in the hidden layer, a GRNN can approximate a continuous function to an arbitrary precision. The GRNN is a memory based network, which provides estimates of continuous variables and converges to the underlying optimal linear or nonlinear regression surface. The network requires no prior knowledge of a specific functional from between input and output. The appropriate form is expressed as a probability density function that is empirically determined from observed data using Parzen window estimation [11]. For this reason, it works very well with sparse data. The network is a one-pass learning algorithm and can generalize from examples as soon as they are stored. The structure of the Network is depicted in Figure 3.

Let x be a vector random variable of dimension p, and y be a scalar random variable. Then $f(x,y)$ is the joint continuous probability density function of x and y. Let X be a particular value of the random variable x. The conditional mean of y given X (regression of y on X) is given by

$$E[y \mid \mathbf{X}] = \frac{\int_{-\infty}^{\infty} y f(\mathbf{X}, y)\, dy}{\int_{-\infty}^{\infty} f(\mathbf{X}, y)\, dy} \qquad (3)$$

But the probability density function $f(x,y)$ is not known a priori. It may be estimated from a sample of observations of x and y as proposed by Parzen as [9]

$$E[y \mid \mathbf{X}] \cong \frac{\sum_{i=1}^{n} Y^{i} \exp\left(-\dfrac{C_i}{\sigma}\right)}{\sum_{i=1}^{n} \exp\left(-\dfrac{C_i}{\sigma}\right)} \qquad (4)$$

where

$$C_i = \sum_{j=1}^{p} \mid X_j - X_j^{i} \mid \qquad (5)$$

is the city block distance. Note that in (4), σ is the spread parameter of the density estimator, and should

not be confused with the echo-width defined in (2). The estimate (4) can be considered as a weighted average of all the observed values, each being weighted exponentially according to its distance from X. It can be shown that this density estimator used in estimating (3) asymptotically converges to the underlying probability density function $f(x,y)$ at all points (x,y) at which the density function is continuous, provided that the spread parameter $\sigma = \sigma(n)$ is chosen as a decreasing function of n. When σ is large, the estimated density function approaches a multivariate Gaussian function. For intermediate values of σ, all values of Y^{i} are taken into account, but those corresponding to points closer to $\mathbf{X}$ are weighted heavier. The estimate cannot converge to poor solutions corresponding to local minima of the error criterion.

TRAINING

The sensors are assumed to be fixed with respect to the wave direction. The target is impinged upon by transverse magnetic plane waves from different directions. To prepare the training data, a total of 10 equally spaced receivers are used.

It was found that the FFT modulus of the echo-width patterns sampled at the receiver positions for angles of incidence provided better generalization capabilities for the network, compared with the case when the network was trained with the echo-width vector (amplitude and phase). Simulated bistatic echo-width was used for the training of the network. The forward problem was solved using the method of moments. These calculations formed a 10 element input vector at every receiver for the network.

Some noisy data created by displacing the receivers, were added to the training data set to let the system face small sensor position drifts. These vectors were used in training the network. The spread parameter σ was manipulated so that the network angular estimation was sufficiently robust. The target value to be trained was the angle between the target orientation and the incident wave.

RESULTS

In this section, the performance of the network will be examined.

The network was trained using the data described in the previous section for the triangular shaped target shown in Figure 1. To check the generalization power

of the network, a set of 40 new input data was produced, this time by angles not previously encountered by the network with all other parameters held unchanged. Figure 4 shows the cumulative error for estimating the target orientation for trained and untrained data. The network clearly displays a very good level of generalization of its estimates based on the training data set and more than 99% of the cases have less than one-degree error.

The triangular cylindrical target shown in Figure 5 was considered next. This is the Ipswich target IPS-009. The network was trained using the simulated echo-width data. Then the network was presented with the RCS data collected at Ipswich bistatic RCS range. Only the bistatic echo-width data on 180 degree range was used (that is, one side of the cylinder was examined). The performance of the network in estimating the orientation of the target is shown in Figure 6. It is observed that the error is less than one degree in more than 98% of the cases.

Next, the performance of the network was examined for the case when the target size is not exactly known a priori, but rather the geometry of its shape is known. The system was tested in facing an elliptical cylinder when the electrical size of the cross-section was rescaled from −7% to +5%. The cumulative error is shown in Figure 7 for three different frequency scaling factors in the above range.

CONCLUDING REMARKS

The problem of estimation of two-dimensional conducting target orientation was efficiently handled by a Generalized Regression Neural Network. The training data set consisted of the calculated bistatic echo-width data when the target was exposed by an incident single frequency TM plane wave. The performance of the network does not change if the frequency of the plane wave is altered. Currently, The network performance against sensor misplacements, sensor noise (correlated and uncorrelated), are under study.

It is believed that time domain schemes such as range profiling techniques can utilize this method to overcome difficulties in estimating the orientation of the target.

ACKNOWLEDGEMENTS

REFERENCES

[1] L. Marquez and T. Hill, "Function approximation using back-propagation and general regression neural networks," *Proceeding of the Twenty-Sixth Hawaii International Conference on System Sciences*, Vol. 4, pp. 607-615, 1993.

[2] S. P. Jacobs and J. A. O'Sullivan, "High-resolution radar models for joint tracking and recognition," *IEEE National Radar Conference*, pp. 99 –104, 1997.

[3] S. P. Jacobs and J. A. O'Sullivan, "Automatic target recognition using sequences of high-resolution radar range-profiles ," *IEEE Trans. Aerospace & Electronic Systems*, Vol. 36, No. 2, April 2000.

[4] H. J. Li and S. H. Yang, "Using range profiles as feature vectors to identify aerospace objects," *IEEE Trans. Antennas & Propagat.*, Vol. 41, No. 3, pp. 261-268, March 1993.

[5] H. J. Lie and C. H. Ahn, "New iterative inverse scattering algorithms based on neural networks," *IEEE Trans. Magnetics*, Vol. 30, No. 5, September 1994.

[6] I. Elshafiey, L. Udpa, and S. S. Upda, "Application of neural networks to inverse problems in electromagnetics," *IEEE Trans. Magnetics*, Vol. 30, No. 5, September 1994.

[7] S.Caorsi and P.Gamba, "Detection of dielectric cylinder by a neural network approach," *IEEE Trans. Geoscienece & Remote Sensing*, Vol. 37, pp. 820-827, 1999.

[8] S.Chakrabarti, N. Bindal, and K.Theagharajan, "Robust radar target classifier using artificial neural networks*," IEEE Trans. Neural Networks*, Vol. 6, No. 3, May 1995.

[9] D. F. Specht, "A general regression neural network," *IEEE Trans. Neural Networks*, Vol. 2, No. 6, pp. 568 –576, November 1991.

[10] A. H. El Zooghby, C. G. Christodoulou, and M. Georgiopoulos, "Performance of radial-basis function networks for direction of arrival estimation with antenna arrays," *IEEE Trans. Antennas & Propagat.*, Vol. 45, No. 11, November 1997.

[11] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, Vol. 33, pp. 1065-1076, 1962.

Ali Kabiri was born in September 1978 in Tehran, Iran. He received his B.S. degree in Electrical Engineering from Sharif University of Technology in the year 2000. Mr. Kabiri received his M.S. degree in Physics from Tehran University in the year 2002. He is currently active in the design and implementation of automated-based systems.

Nima Sarshar was born in 1978 in Shiraz, Iran. He received the B.S. degree from Sharif University of Technology, Iran, in 2000, and the M.S. degree from University of California, Los Angeles in 2002 both in Electrical Engineering. Mr. Sarshar is currently working towards his PhD degree at McMaster University, Complex Systems Laboratory. His research interests include statistical inference, inverse problems, physical models of complex networks, and large scale communication networks.

Kasra Barkeshli was born on August 12, 1961 in Tehran, Iran. He received his B.S. and M.S. degrees from the University of Kansas in 1982 and 1984, and the Ph.D. degree from The University of Michigan in 1991 all in electrical engineering. He also holds a M.S. degree in Mathematics from The University of Michigan.

Dr. Barkeshli is an Associate Professor of Electrical Engineering at Sharif University of Technology in Tehran. He was the Chairman of the Electrical Engineering Department from 1995 to 1999. Since 1999, he has been the Vice Chancellor for Student Affairs at Sharif University of Technology. His primary research interests include antennas theory, inverse scattering, and computational methods in electromagnetics.
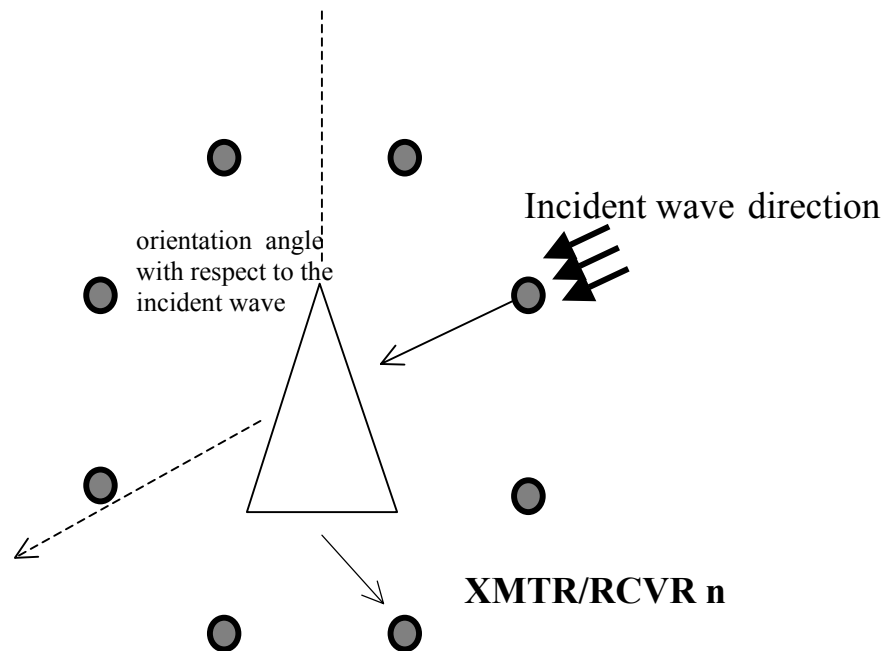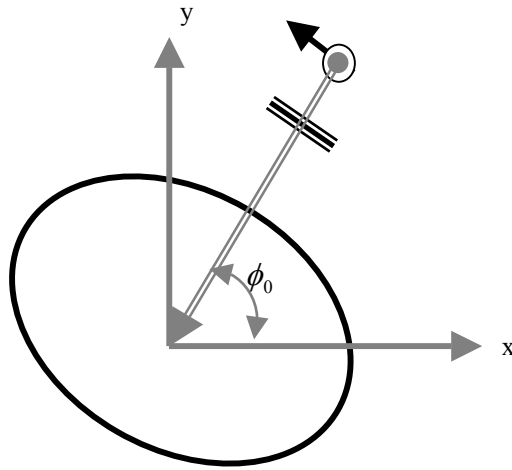
Figure 1- Problem set-up.

Figure 2- A uniform plane wave impinging upon a perfectly conducting cylinder.

$$a_1=\text{radbas}(\|\mathbf{IW}_{1,1}-\mathbf{P}\|\mathbf{b}_1\mathbf{I}) \qquad a_2=\text{purelin}(n_2)$$
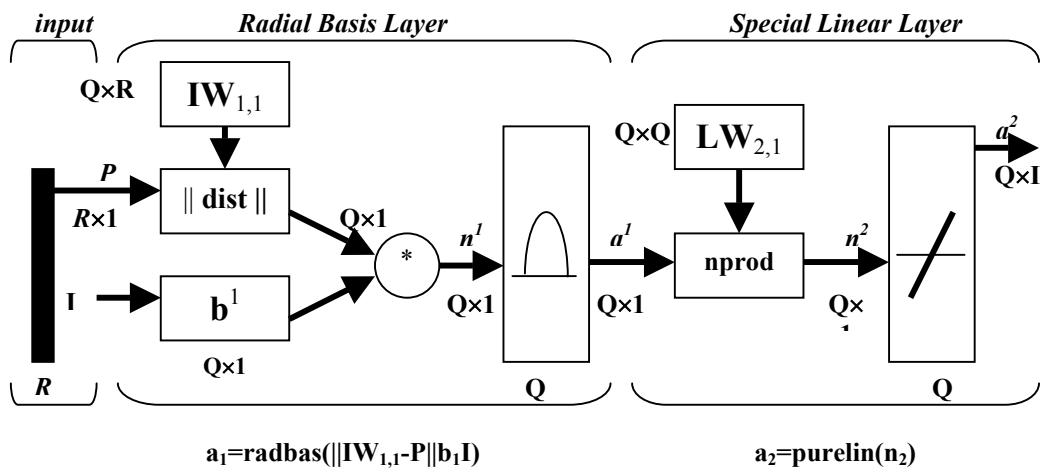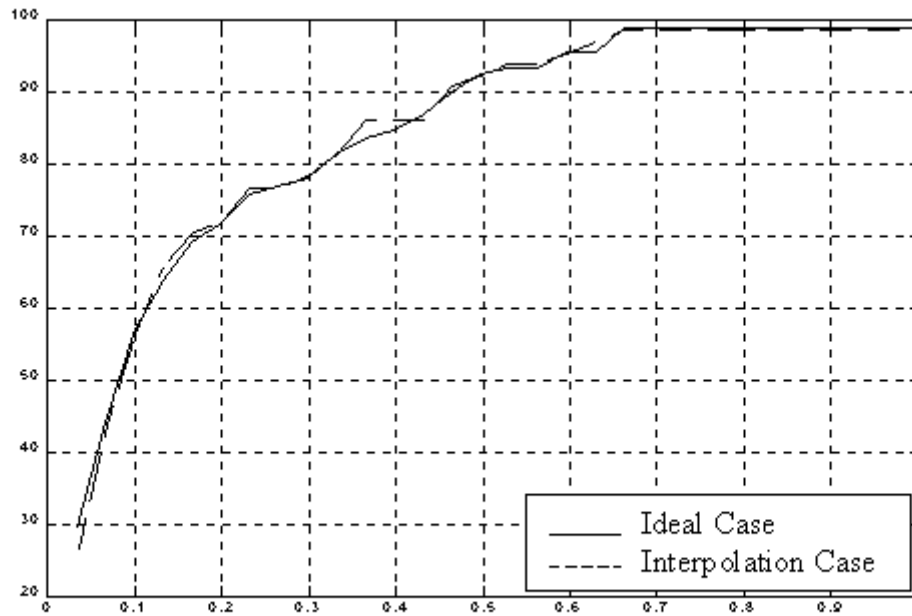
Figure 3- The structure of GRNN.

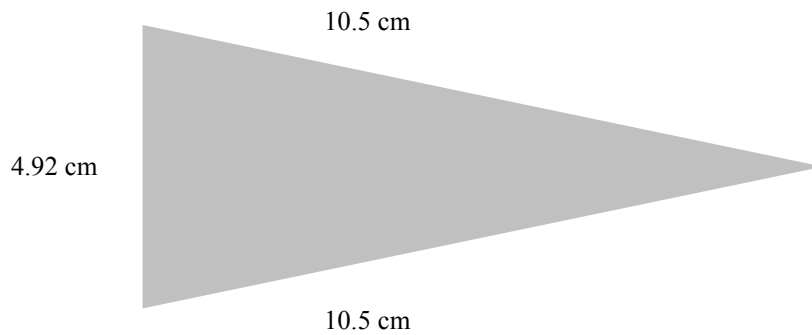Figure 4- The GRNN estimates the orientation of the target shown
in Figure 1 by the concept of generalization.

Figure 5- A triangular cylinder of sides $10.5cm \times 4.92cm \times 10.5cm$ illuminated
by a 10 GHz TM^z plane wave. The height of the cylinder is 40.8 cm.
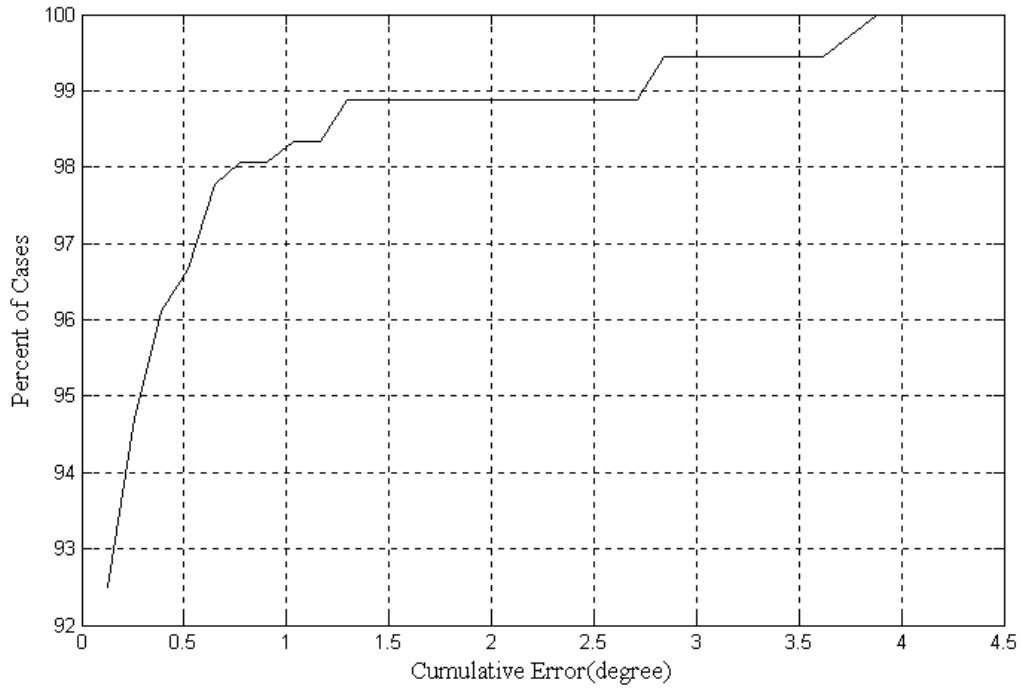
Figure 6- Error diagram for the network response for the target shown in Figure 5.

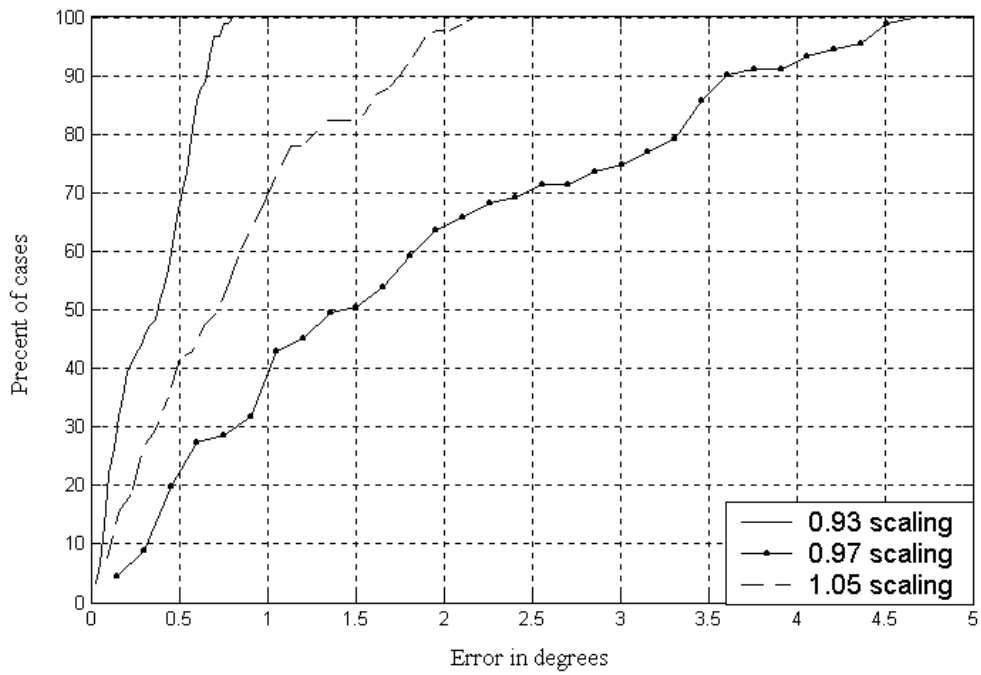
Figure 7- Cumulative error at various levels of frequency scaling.

Application of Two-Dimensional AWE Algorithm in Training Multi-Dimensional Neural Network Model

Y. Xiong, D. G. Fang, and R. S. Chen

School of Electronic and Photoelectric Technology

Nanjing University of Science and technology,

Nanjing Jiangsu 210094, P. R. China

ABSTRACT

Artificial neural network (ANN) plays very important role in microwave engineering. Training a neural network model is the key of neural network technique. The conventional methods for training, such as method of moment (MoM), are time-consuming when the training parameters are a bit more. In order to aid the training process by reducing the amount of costly and time-consuming sampling cycles, a lot of algorithms have been developed, such as asymptotic waveform evaluation (AWE). In this paper, MoM in conjunction with the two-dimensional AWE is applied to accelerate the process of training the neural network model based on the input impedance response on frequency and that on other parameters of a microstrip antenna. In AWE method, the derivatives of Green's function are required. A closed form of microstrip Green's function is used for this requirement. Then, the derivative matrices respect to both frequency and permittivity can be obtained from the original matrix. With these matrices in hand, coefficients of the two-dimensional Pade polynomial can be obtained. So the sampling data for training neural network model can be obtained and the process of training neural net model can be completed quickly and accurately. Numerical results demonstrate the efficiency of this technique.

KEY TERMS

AWE, neural network, microstrip antennas

1 INTRODUCTION

Artificial neural networks (ANNs) have emerged as a powerful technique for modeling general input/output relationships. ANNs provide electromagnetically trained ANN (EM-ANN) models for use in CAD of RR/microwave circuits, antennas, and systems [1]. The training is the most important step in the development of ANNs. The actual training process involves algorithms for finding values of weights associated with various neurons. This process can be viewed as an optimization one. Various well-known optimization techniques, such as genetic algorithms and so on, can be used for this purpose. This process is quite time-consuming. For example, to train an ANN which is available in a wide frequency band, the computation should be carried out repeatedly at different frequencies. To overcome this difficulty, the space-mapping (SM) technique has been introduced [2]. This technique establishes a mathematical link between the coarse and the fine models and directs the bulk of the CPU-intensive computation to the coarse model, while preserving the accuracy offered by the fine model. Alternatively, the asymptotic waveform evaluation (AWE) has also been applied in finite difference solution [3-6]. This technique extrapolates the data from one point to a certain range based on the value and the high order derivatives at this point. From this concept, it is seen that this technique is computationally efficient due to involving the analytical relationships and is available to the cases where the derivatives may be obtained. AWE requires the derivatives of Green's functions, so it is often

used for free-space problems [7-8]. In this paper, the 2-D AWE has been developed to extrapolate the responses over frequency and permittivity simultaneously to characterize microstrip antennas, so the response over certain frequency and permittivity ranges can be extrapolated from single point accurately and quickly. To check the validity of this method, the analysis of microstrip patch antenna is chosen as an example by using method of moments (MoM). The variables in the model are frequency, relative permittivity, position of feed line and the dimension of the patch. In the training process, two-dimensional AWE is responsible for providing the response of both frequency and relative permittivity simultaneously within certain range.

2   FORMULAS

2.1   Two dimensional AWE method [9]

MoM with the substrate Green's function usually results in a matrix equation in the following form:

$$Z(k, \varepsilon_r) I(k, \varepsilon_r) = V(k, \varepsilon_r) \qquad (2.1)$$

Where Z is a square matrix and only can be determined by the object analyzed, I is an unknown vector of the induced currents on the patch, V is a known vector associated with the source or excitation, and k is the wave number and ε_r is permittivity. In accordance with the AWE method, $I(k, \varepsilon_r)$ is expanded into a two-dimensional Taylor series to obtain the solutions of (2.1) over certain frequency and permittivity ranges.

$$I(k, \varepsilon_{r0}) = \sum_{n=0}^{Q} \sum_{m=0}^{P} a_{nm} (k - k_0)^n (\varepsilon_r - \varepsilon_{r0})^m$$

$$\qquad (2.2)$$

$$a_{nm} = Z^{-1} \left[\frac{1}{(n+m)!} C_{n+m}^n \frac{\partial^{m+n} V}{\partial^n k \partial^m \varepsilon_r} - \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} a_{ij} \frac{1}{(n+m-i-j)!} \times \right. $$

$$\left. C_{n+m-i-j}^{n-i} \frac{\partial^{n+m-i-j} Z}{\partial^{n-i} k \partial^{m-j} \varepsilon_r} \frac{1}{(n+m-i-j)!} \right]$$

$$\qquad (2.3)$$

Where k_0 is the wave number on the expansion point, a_{nm} denote the unknown coefficients, and $P \times Q$ denotes the total number of such coefficients.

In order to get the coefficients a_{nm}, the derivatives of matrix I have to be generated.   A closed form Green's function $G_e^{(1)}(\rho)$ that is easy to get derivatives is used [10].

$$G_e^{(1)}(\rho) = a \frac{\pi}{2} [L_0(\frac{\varepsilon_r \rho}{h}) - L_0(\frac{\rho}{\mu_r h})]$$

$$- a^2 (\frac{e^{-jk_0\rho} - 1}{k_0 \rho}) \qquad (2.4)$$

$$L_0(z) = H_0(z) - Y_0(z) \qquad (2.5)$$

Where a is defined as $a = ((\varepsilon_r \mu_r - 1) / \varepsilon_r) k_0 h$, H_0 and Y_0 are the Struve and Neumann functions of zero order and $\rho = \sqrt{(x-x')^2 + (y-y')^2}$. The Green's function $G_e^{(1)}(\rho)$ in (2.4) is valid subject to the conditions $k_0 \rho (k_0 h / \varepsilon_r)^2 << 1$ and $k_0 \rho (\mu_r k_0 h)^2 << 1$.

The Taylor expansion has a limited bandwidth. To obtain a wider bandwidth, we represent $I(k, \varepsilon_r)$ with a better rational Padé function:

$$I(k, \varepsilon_r) = \frac{\sum_{i=0}^{X} \sum_{j=0}^{Y} b_{ij} (k - k_0)^i (\varepsilon_r - \varepsilon_{r0})^j}{\sum_{l=0}^{F} \sum_{m=0}^{G} c_{lm} (k - k_0)^l (\varepsilon_r - \varepsilon_{r0})^m}$$

$$\qquad (2.6)$$

Where $c_{00} = 1$ and XF+YG+X+F+Y+G+1 =PQ+P+Q. If we make Y=G, the unknown coefficients $b_{i,j}$ and $c_{i,j}$ can be calculated by substituting (2.2) into (2.6), multiplying (2.6) by the denominator of the Padé expansion, and matching the coefficients of the equal powers of k-k_0 and $\varepsilon_r - \varepsilon_{r0}$. This leads to the matrix equation (2.7). Where n is from 1 to X. If we solve equation (2.7) in turn, $b_{i,j}$ and $c_{i,j}$ can be obtained,

and the current vector $I(k, \varepsilon_r)$ can be obtained by the calculated Padé model.

2.2 Neural networks [11]

Multilayer perceptrons (MLP) are the most popular type of neural networks in use today. Typically, an MLP neural network consists of an input layer, one or more hidden layer, and an output layer, as shown in Fig. 2.1. The top layer is the output layer and the input impedance and other scattering parameters can be outputted. The bottom layer is the input layer, and four parameters, frequency, relative permittivity, position of feed line and the dimension of patch, are inputted. The other two layers are hidden layers, and it can be automatic treated in the software [12].

can be simultaneously obtained by the two-dimensional AWE method. In AWE, the differentiation operates on the Green's function which does not involve the dimensions of the object to be analyzed. Therefore it is not available to obtain the response with respect to the dimensions through AWE. In this case, the sampling data for training variables respect to the dimension of the microstrip and position of feed line can only be calculated point by point. Even in this case, the speed of training is about one or two orders faster than that of direct training. With the two-dimensional AWE method and neural network technique in hand, we can accurately and efficiently construct the neural network model. The flowchart is shown in Fig. 2.2.

$$
\begin{bmatrix}
1 & 0 & \cdots & 0 & 0 & \cdots & 0 & -a_{0,0} \\
0 & 1 & \cdots & 0 & 0 & \cdots & -a_{0,0} & -a_{1,0} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 1 & -a_{X-F,0} & \cdots & -a_{X-1,0} & -a_{X,0} \\
0 & 0 & \cdots & 0 & -a_{X-F+1,0} & \cdots & -a_{X,0} & -a_{X+1,0} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & -a_{X+1,0} & \cdots & -a_{X+F,0} & -a_{X+F+1,0}
\end{bmatrix}
\begin{bmatrix}
b_{0,n} \\
b_{1,n} \\
\vdots \\
b_{X,n} \\
c_{F,n} \\
\vdots \\
c_{0,n}
\end{bmatrix}
$$

$$
=
\begin{bmatrix}
\sum_{i=0}^{n-1} c_{0,i} a_{0,n-i} \\
\sum_{i=0}^{n-1} c_{0,i} a_{1,n-i} + \sum_{i=0}^{n-1} c_{1,i} a_{0,n-i} \\
\vdots \\
\sum_{i=0}^{n-1} \sum_{j=0}^{X} c_{j,i} a_{X-j,n-i} \\
\sum_{i=0}^{n-1} \sum_{j=0}^{X+1} c_{j,i} a_{X+1-j,n-i} \\
\vdots \\
\sum_{i=0}^{n-1} \sum_{j=0}^{X+F+1} c_{j,i} a_{X+F+1-j,n-i}
\end{bmatrix}
$$

2.3 Hybrid of AWE and ANN

AWE method is an accurate and efficient technique that is based on the electromagnetic mechanism. In the practical application, the response varied with frequency and permittivity

As the neural network model is constructed, the response of object varied with each parameter can be immediately obtained. This trained model may be used in the optimization of microstrip structures other than microstrip antennas.
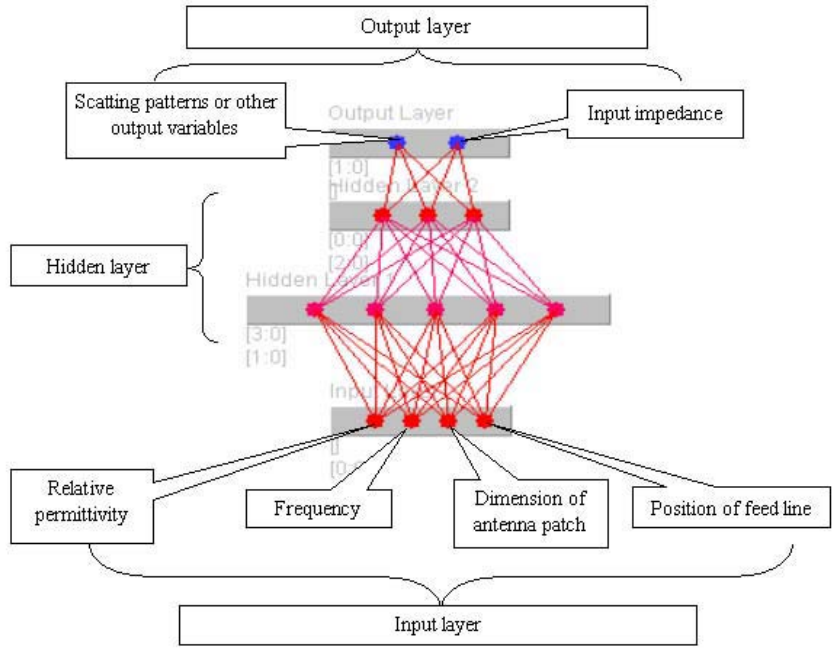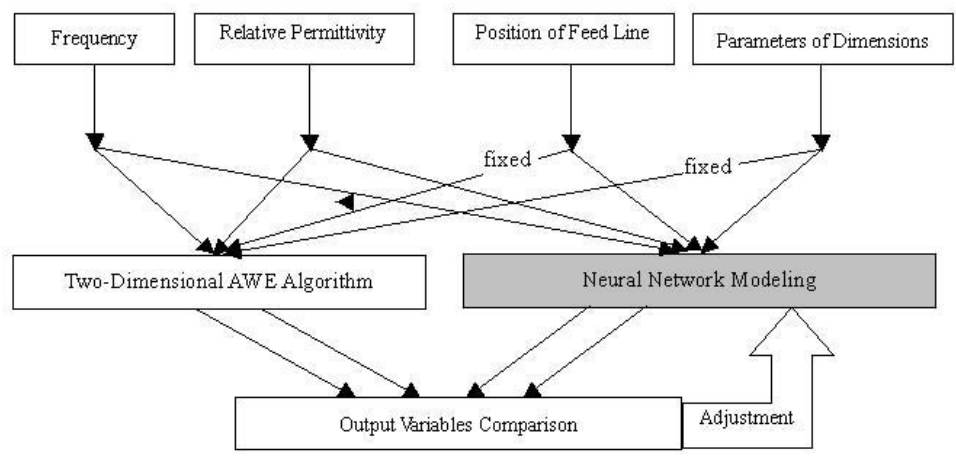
Figure 2.1 Multilayer perceptrons (MLP) structure.

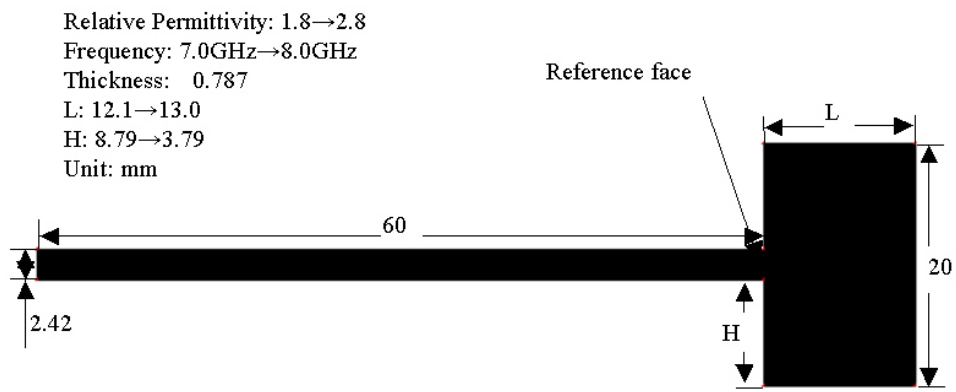Figure 2.2 The process of the hybrid technique.

Relative Permittivity: 1.8→2.8
Frequency: 7.0GHz→8.0GHz
Thickness: 0.787
L: 12.1→13.0
H: 8.79→3.79
Unit: mm

Fig.3.1 The antenna geometry

3   NUMERICAL RESULTS AND DISCUSSION

The example is a microstrip antenna consisting of a conducting patch residing on a dielectric substrate having thickness h=0.787mm (Fig. 3.1). The increments of frequency, relative permittivity, position of feed line H and dimension of patch L are 0.01GHz, 0.01, 0.1mm and 0.01mm respectively. In order to get the response under the following specification: the frequency varies from 7.5GHz to 8.4GHz; the permittivity varies from 1.8 to 2.8; the dimension L varies from 12.0mm to 13.0mm; the feed line position H varies from 8.79 to 0.79, the direct method requires 7.5×10^9 seconds to obtain the solution on a Personal Computer (1.2GHz AMD K7 processor). With general neural network algorithm, including the training time, to obtain the same accuracy, 1.4×10^6 seconds are need. But with hybrid method, only 1.2×10^5 seconds are needed, which is 6.3×10^4 times faster than the direct method and 1.2×10^1 times faster than the general neural network method (Table 3.1). The training process of the software-"Neuralmodeler" is shown in Fig. 3.2 and the final error is less than 0.01. Figure 3.3 and Figure 3.4 show the real and imaginary parts of the input impedance as a function of frequency, relative permittivity, dimension L and position H by using the hybrid method of the two dimensional AWE method and neural network algorithm, respectively (Due to difficulty in presenting four dimensional figure, the variables, dimension L and position H, are fixed). When this four variables neural network model for this antenna patch is obtained, consequently the optimizing

and designing will be an easy job. This neural network model gives the complete characterization of the microstrip patch antenna. Because of its computational efficiency, it is realizable in the optimization and the observation of the sensitivity of the parameters, such as the relative permittivity (Fig. 3.5). In this example, only four variables are involved. It is observed that the more the variables to be optimized, the more the reduction of the computer time.

4   CONCLUSION

The AWE algorithm has been extended from one dimensional to two dimensional cases. This extension results in the extrapolation for two variables simultaneously. Compared to the one-dimensional AWE, the computer time is further reduced significantly. The hybrid of AWE and ANN makes full use of the advantages of both algorithms. Numerical results demonstrate the efficiency of this hybrid scheme.

Table 3.1 The compare of neural network methods and direct method

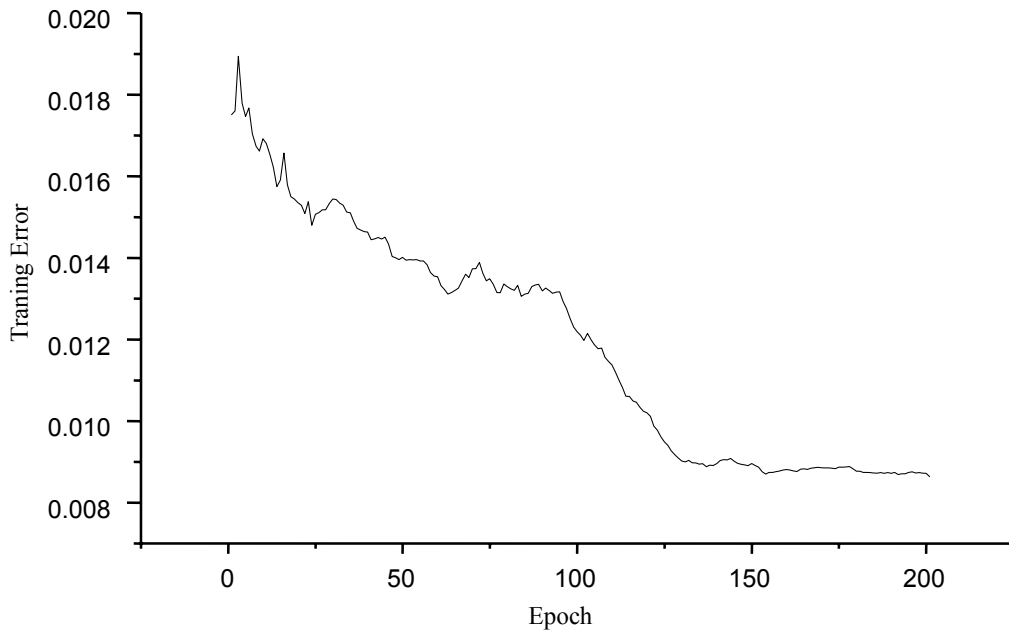| Time		Neural network method		Direct method
		Hybrid method	General method	
Training	Sampling	1.2×10^5 s	1.4×10^6 s	No training
	Modeling	7.0×10^1 s	7.0×10^1 s	
Generating response		Almost zero	Almost zero	7.5×10^9 s
Total time		1.2×10^5 s	1.4×10^6 s	7.5×10^9 s

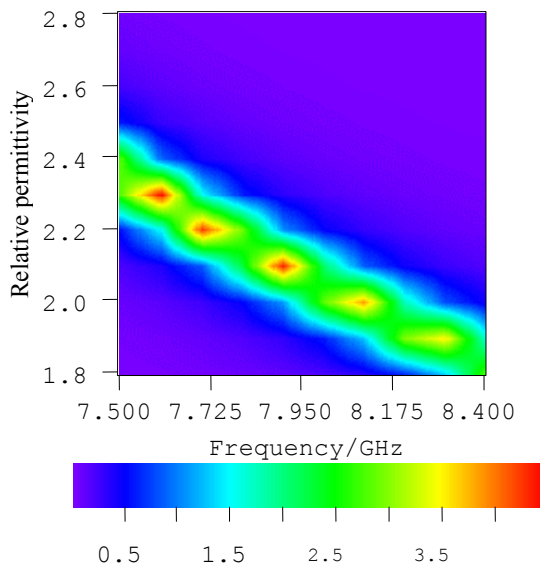Figure 3.2 Effect of hybrid method on the training errors.

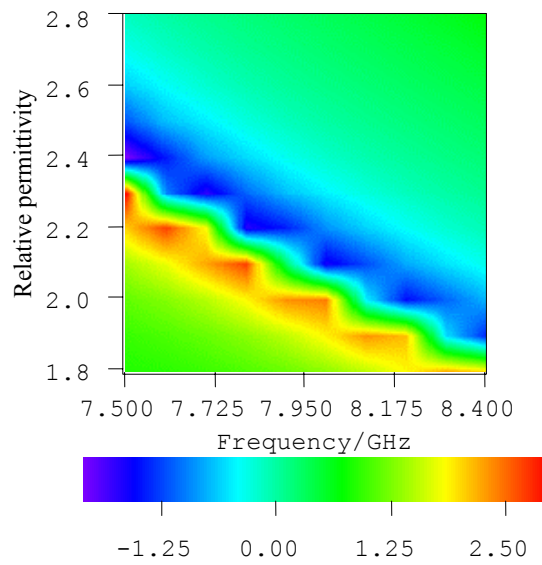Figure 3.3 The real part of the input impedance (L =12.5 mm, H=8.79mm).

Figure 3.4 The imaginary part of the input impedance (L =12.5 mm, H =8.79mm).
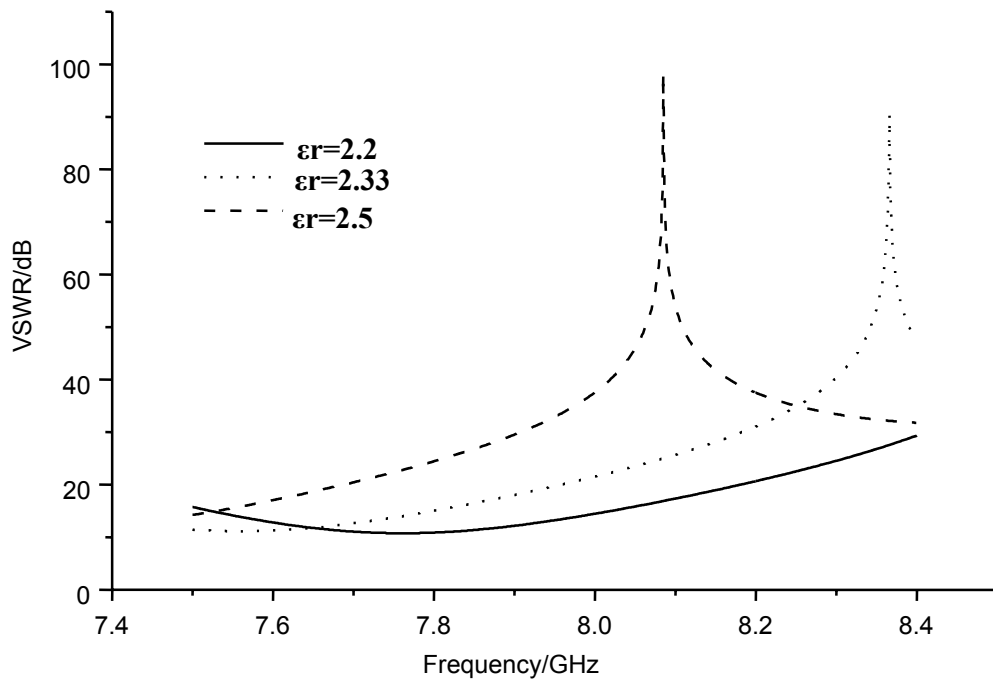
Figure 3.5 Sensitivity of the permittivity (L=12.5mm, H=8.79mm).

5 REFERENCES

[1] V. Devabhaktuni, M. C. E.Yagoub, Y. Fang, J. Xu and Q. J. Zhang, "Neural networks for microwave modeling: model development issues and nonlinear modeling technique," *International Journal of RF and Microwave CAE*, (invited, to be published).

[2] J. W. Bandler, M. A. Ismail, J. E. Rayas-Sanchez and Q.J.Zhang, "Neuromodeling of microwave circuits exploiting space mapping technology," *IEEE Trans. Microwave Theory Tech.*, vol. 47, pp. 2417-2427, December 1999.

[3] E. K. Miller, "Model-based parameter estimation in electromagnetic Pt.1," *IEEE Antennas and Propagation Magazine*, vol. 40, no. 1, pp. 40-52, 1998.

[4] E. K. Miller, "Model-based parameter estimation in electromagnetic Pt. 2," *IEEE Antennas and Propagation Magazine*, vol. 40, no.1, pp. 51-65, 1998.

[5] E. K. Miller, "Model-based parameter estimation in electromagnetic Pt. 3," *IEEE Antennas and Propagation Magazine*, vol.

[6] M. Li, Q. J. Zhang and M.S.Nakhla, "Finite difference solution of EM fields by asymptotic waveform techniques," *IEE Proceedings on Microwaves, Antennas and Propagation*, vol. 143, pp. 512-520, 1996

[7] D. Jiao and J. M. Jin, "Asymptotic Waveform Evaluation for Scattering by a Dispersive Dielectric Object," *Microwave Opt. Tech Lett.,* vol. 24, no. 4, pp. 232-234, Feb. 2000

[8] C. M. Tong and W. Hong, "Fast Calculation of Wide Angle Mono-static RCS of Dielectric Cylinders Based on Asymptotic Wave Evaluation Technique," *Chinese Journal of Radio Science*, vol. 16, no. 1, pp. 72-75, 2001

[9] Y. Xiong, D. G. Fang and F. Ling, "Two-Dimensional AWE Technique in Fast Calculation," ICMMT'2002, pp. 393-396

[10] Ahmad Hoorfar, "Simple Closed-Form Expressions for Microstrip Green's Functions in a Magneto-dielectric Substrate," *Mircrowave Opt. Tech. Letters,* vol. 8, no.1, pp. 33-36, Jan. 1995

[11] Q. J. Zhang and K.C.Cupta, *Neural Networks for RF and Microwave Design*, Artech House, 2000.

[12] Q. J. Zhang and his research team, software "*NeuralModeler*", Version 1.2.2 for windows NT 4.0

Xiong Ye was born in JiangSu province in October 1978. He received the bachelor degree in electro-magnetics and microwave technology from Nanjing University of Science and Technology (NUST) in 1996.

At present he is pursing postgraduate degree in millimeter wave technique laboratory in NUST, His main research interests are radar image analysis, radar cross section prediction and AWE technique.

FANG Dagang (SM'90,F'03) was born in Shanghai in June 1937. He graduated from graduate school of Beijing Institute of Post and Telecommunications in 1966. From 1980 to 1982, he was Visiting Scholar at Laval University and University of Waterloo both in Canada. He has been a Professor of Nanjing University of Science and Technology since 1986. He was qualified as Ph.D. student supervisor in 1990. He had been Visiting Professor of six universities in Canada and in Hong Kong since 1987. He is IEEE Fellow and on the editorial board of IEEE Trans. MTT and several other journals. He also serves as Vice Editor of "The Chinese Journal of Microwaves". He has co-edited one proceedings of international conference, authored or co-authored one book, one textbook, two book chapters and over 300 papers including more than 40 papers published in the international journals. His research interests include computational electromagnetics, microstrip antennas, EM scattering and microwave imaging and smart antenna.

Ru-Shan Chen received the B.Sc. degree in 1987 and M.Sc. degree in 1990 both from Dept. of Radio engineering, Southeast University and PhD from Dept of Electronic Engineering, City University of Hong Kong in 2001. He joined the Dept. of Electrical Engineering, Nanjing University of Science & Technology (NUST), where he was a Teaching Assistant in 1990, and Lecture later in 1992. Since Sept 1996, he was a Visiting Scholar in Dept. of Electronic Engineering, City University of Hong Kong, first as RA, became a Senior RA in July 1997 and a RF in April 1998. From June to Sept. 1999, He was also a Visiting Scholar at Montreal University, Canada. In Sept. 1999, he was promoted to be a full Professor and Associate Director of Microwave & Communication Center in NUST and Associate Head, Dept of Communication Engineering, NUST in 2001. He ever received the 1992 third-class science and technology advance prize, the 1993 third-class science and technology advance prize and the 1996 second-class science and technology advance prize given by National Education Committee of China, the 1999 first-class science and technology advance prize given by Jiang Su Province, the 2001 second-class science and technology advance prize. He has authored or co-authored more than 120 papers including 60 paper in international journals. His research interests mainly include microwave/millimeter wave system, measure-ments, antenna, circuit and computational electromagnetics.

ACES COPYRIGHT FORM

This form is intended for original, previously unpublished manuscripts submitted to ACES periodicals and conference publications. The signed form, appropriately completed, MUST ACCOMPANY any paper in order to be published by ACES. PLEASE READ REVERSE SIDE OF THIS FORM FOR FURTHER DETAILS.

TITLE OF PAPER:

RETURN FORM TO:
Dr. Atef Z. Elsherbeni
University of Mississippi
Dept. of Electrical Engineering

AUTHORS(S)
Anderson Hall Box 13
PUBLICATION TITLE/DATE:
University, MS 38677 USA

PART A - COPYRIGHT TRANSFER FORM

(NOTE: Company or other forms may not be substituted for this form. U.S. Government employees whose work is not subject to copyright may so certify by signing Part B below. Authors whose work is subject to Crown Copyright may sign Part C overleaf).

The undersigned, desiring to publish the above paper in a publication of ACES, hereby transfer their copyrights in the above paper to The Applied Computational Electromagnetics Society (ACES). The undersigned hereby represents and warrants that the paper is original and that he/she is the author of the paper or otherwise has the power and authority to make and execute this assignment.

Returned Rights: In return for these rights, ACES hereby grants to the above authors, and the employers for whom the work was performed, royalty-free permission to:
1. Retain all proprietary rights other than copyright, such as patent rights.
2. Reuse all or portions of the above paper in other works.
3. Reproduce, or have reproduced, the above paper for the author's personal use or for internal company use provided that (a) the source and ACES copyright are indicated, (b) the copies are not used in a way that implies ACES endorsement of a product or service of an employer, and (c) the copies per se are not offered for sale.
4. Make limited distribution of all or portions of the above paper prior to publication.
5. In the case of work performed under U.S. Government contract, ACES grants the U.S. Government royalty-free permission to reproduce all or portions of the above paper, and to authorize others to do so, for U.S. Government purposes only.

ACES Obligations: In exercising its rights under copyright, ACES will make all reasonable efforts to act in the interests of the authors and employers as well as in its own interest. In particular, ACES REQUIRES that:
1. The consent of the first-named author be sought as a condition in granting re-publication permission to others.
2. The consent of the undersigned employer be obtained as a condition in granting permission to others to reuse all or portions of the paper for promotion or marketing purposes.

In the event the above paper is not accepted and published by ACES or is withdrawn by the author(s) before acceptance by ACES, this agreement becomes null and void.

AUTHORIZED SIGNATURE   TITLE (IF NOT AUTHOR)

EMPLOYER FOR WHOM WORK WAS PERFORMED   DATE FORM SIGNED

Part B - U.S. GOVERNMENT EMPLOYEE CERTIFICATION

(NOTE: if your work was performed under Government contract but you are not a Government employee, sign transfer form above and see item 5 under Returned Rights).

This certifies that all authors of the above paper are employees of the U.S. Government and performed this work as part of their employment and that the paper is therefor not subject to U.S. copyright protection.

AUTHORIZED SIGNATURE   TITLE (IF NOT AUTHOR)

NAME OF GOVERNMENT ORGANIZATION   DATE FORM SIGNED

PART C - CROWN COPYRIGHT

(NOTE: ACES recognizes and will honor Crown Copyright as it does U.S. Copyright. It is understood that, in asserting Crown Copyright, ACES in no way diminishes its rights as publisher. Sign only if *ALL* authors are subject to Crown Copyright).

This certifies that all authors of the above Paper are subject to Crown Copyright. (Appropriate documentation and instructions regarding form of Crown Copyright notice may be attached).

_____

AUTHORIZED SIGNATURE   TITLE OF SIGNEE

_____

NAME OF GOVERNMENT BRANCH   DATE FORM SIGNED

Information to Authors

ACES POLICY

ACES distributes its technical publications throughout the world, and it may be necessary to translate and abstract its publications, and articles contained therein, for inclusion in various compendiums and similar publications, etc. When an article is submitted for publication by ACES, acceptance of the article implies that ACES has the rights to do all of the things it normally does with such an article.

In connection with its publishing activities, it is the policy of ACES to own the copyrights in its technical publications, and to the contributions contained therein, in order to protect the interests of ACES, its authors and their employers, and at the same time to facilitate the appropriate re-use of this material by others.

The new United States copyright law requires that the transfer of copyrights in each contribution from the author to ACES be confirmed in writing. It is therefore necessary that you execute either Part A-Copyright Transfer Form or Part B-U.S. Government Employee Certification or Part C-Crown Copyright on this sheet and return it to the Managing Editor (or person who supplied this sheet) as promptly as possible.

CLEARANCE OF PAPERS

ACES must of necessity assume that materials presented at its meetings or submitted to its publications is properly available for general dissemination to the audiences these activities are organized to serve. It is the responsibility of the authors, not ACES, to determine whether disclosure of their material requires the prior consent of other parties and if so, to obtain it. Furthermore, ACES must assume that, if an author uses within his/her article previously published and/or copyrighted material that permission has been obtained for such use and that any required credit lines, copyright notices, etc. are duly noted.

AUTHOR/COMPANY RIGHTS

If you are employed and you prepared your paper as a part of your job, the rights to your paper initially rest with your employer. In that case, when you sign the copyright form, we assume you are authorized to do so by your employer and that your employer has consented to all of the terms and conditions of this form. If not, it should be signed by someone so authorized.

        NOTE RE RETURNED RIGHTS: Just as ACES now requires a signed copyright transfer form in order to do "business as usual", it is the intent of this form to return rights to the author and employer so that they too may do "business as usual". If further clarification is required, please contact: The Managing Editor, R. W. Adler, Naval Postgraduate School, Code EC/AB, Monterey, CA, 93943, USA (408)656-2352.

Please note that, although authors are permitted to re-use all or portions of their ACES copyrighted material in other works, this does not include granting third party requests for reprinting, republishing, or other types of re-use.

JOINT AUTHORSHIP

For jointly authored papers, only one signature is required, but we assume all authors have been advised and have consented to the terms of this form.

U.S. GOVERNMENT EMPLOYEES

Authors who are U.S. Government employees are not required to sign the Copyright Transfer Form (Part A), but any co-authors outside the Government are.

Part B of the form is to be used instead of Part A only if all authors are U.S. Government employees and prepared the paper as part of their job.

        NOTE RE GOVERNMENT CONTRACT WORK: Authors whose work was performed under a U.S. Government contract but who are not Government employees are required so sign Part A-Copyright Transfer Form. However, item 5 of the form returns reproduction rights to the U. S. Government when required, even though ACES copyright policy is in effect with respect to the reuse of material by the general public.
January 2002

INFORMATION FOR AUTHORS

PUBLICATION CRITERIA

Each paper is required to manifest some relation to applied computational electromagnetics. **Papers may address general issues in applied computational electromagnetics, or they may focus on specific applications, techniques, codes, or computational issues.** While the following list is not exhaustive, each paper will generally relate to at least one of these areas:

1. **Code validation.** This is done using internal checks or experimental, analytical or other computational data. Measured data of potential utility to code validation efforts will also be considered for publication.

2. **Code performance analysis.** This usually involves identification of numerical accuracy or other limitations, solution convergence, numerical and physical modeling error, and parameter tradeoffs. However, it is also permissible to address issues such as ease-of-use, set-up time, run time, special outputs, or other special features.

3. **Computational studies of basic physics.** This involves using a code, algorithm, or computational technique to simulate reality in such a way that better, or new physical insight or understanding, is achieved.

4. **New computational techniques**, or new applications for existing computational techniques or codes.

5. **"Tricks of the trade"** in selecting and applying codes and techniques.

6. **New codes, algorithms, code enhancement, and code fixes.** This category is self-explanatory, but includes significant changes to existing codes, such as applicability extensions, algorithm optimization, problem correction, limitation removal, or other performance improvement. **Note: Code (or algorithm) capability descriptions are not acceptable, unless they contain sufficient technical material to justify consideration.**

7. **Code input/output issues.** This normally involves innovations in input (such as input geometry standardization, automatic mesh generation, or computer-aided design) or in output (whether it be tabular, graphical, statistical, Fourier-transformed, or otherwise signal-processed). Material dealing with input/output database management, output interpretation, or other input/output issues will also be considered for publication.

8. **Computer hardware issues.** This is the category for analysis of hardware capabilities and limitations of various types of electromagnetics computational requirements. Vector and parallel computational techniques and implementation are of particular interest.

Applications of interest include, but are not limited to, antennas (and their electromagnetic environments), networks, static fields, radar cross section, shielding, radiation hazards, biological effects, electromagnetic pulse (EMP), electromagnetic interference (EMI), electromagnetic compatibility (EMC), power transmission, charge transport, dielectric, magnetic and nonlinear materials, microwave components, MEMS technology, MMIC technology, remote sensing and geometrical and physical optics, radar and communications systems, fiber optics, plasmas, particle accelerators, generators and motors, electromagnetic wave propagation, non-destructive evaluation, eddy currents, and inverse scattering.

Techniques of interest include frequency-domain and time-domain techniques, integral equation and differential equation techniques, diffraction theories, physical optics, moment methods, finite differences and finite element techniques, modal expansions, perturbation methods, and hybrid methods. This list is not exhaustive.

A unique feature of the Journal is the publication of unsuccessful efforts in applied computational electromagnetics. Publication of such material provides a means to discuss problem areas in electromagnetic modeling. Material representing an unsuccessful application or negative results in computational electromgnetics will be considered for publication only if a reasonable expectation of success (and a reasonable effort) are reflected. Moreover, such material must represent a problem area of potential interest to the ACES membership.

Where possible and appropriate, authors are required to provide statements of quantitative accuracy for measured and/or computed data. This issue is discussed in "Accuracy & Publication: Requiring, quantitative accuracy statements to accompany data," by E. K. Miller, *ACES Newsletter,* Vol. 9, No. 3, pp. 23-29, 1994, ISBN 1056-9170.

EDITORIAL REVIEW

In order to ensure an appropriate level of quality control, papers are peer reviewed. They are reviewed both for technical correctness and for adherence to the listed guidelines regarding information content.

JOURNAL CAMERA-READY SUBMISSION DATES

March issue	deadline 8 January
July issue	deadline 20 May
November issue	deadline 20 September

Uploading an acceptable camera-ready article after the deadlines will result in a delay in publishing this article.

STYLE FOR CAMERA-READY COPY

The ACES Journal is flexible, within reason, in regard to style. However, certain requirements are in effect:

1. The paper title should NOT be placed on a separate page. The title, author(s), abstract, and (space permitting) beginning of the paper itself should all be on the first page. The title, author(s), and author affiliations should be centered (center-justified) on the first page.

2. An abstract is REQUIRED. The abstract should be a brief summary of the work described in the paper. It should state the computer codes, computational techniques, and applications discussed in the paper (as applicable) and should otherwise be usable by technical abstracting and indexing services.

3. Either British English or American English spellings may be used, provided that each word is spelled consistently throughout the paper.

4. Any commonly-accepted format for referencing is permitted, provided that internal consistency of format is maintained. As a guideline for authors who have no other preference, we recommend that references be given by author(s) name and year in the body of the paper (with alphabetical listing of all references at the end of the paper). Titles of Journals, monographs, and similar publications should be in italic font or should be underlined. Titles of papers or articles should be in quotation marks.

5. Internal consistency shall also be maintained for other elements of style, such as equation numbering. As a guideline for authors who have no other preference, we suggest that equation numbers be placed in parentheses at the right column margin.

6. The intent and meaning of all text must be clear. For authors who are NOT masters of the English language, the ACES Editorial Staff will provide assistance with grammar (subject to clarity of intent and meaning).

7. Unused space should be minimized. Sections and subsections should not normally begin on a new page.

PAPER FORMAT

The preferred format for initial submission and camera-ready manuscripts is 12 point Times Roman font, single line spacing and double column format, similar to that used here, with top, bottom, left, and right 1 inch margins. Manuscripts should be prepared on standard 8.5x11 inch paper.

Only camera-ready electronic files are accepted for publication. The term **"camera-ready" means that the material is neat, legible, and reproducible.** Full details can be found on ACES site, Journal section.

ACES reserves the right to edit any uploaded material, however, this is not generally done. It is the author(s) responsibility to provide acceptable camera-ready pdf files. Incompatible or incomplete pdf files will not be processed, and authors will be requested to re-upload a revised acceptable version.

SUBMITTAL PROCEDURE

All submissions should be uploaded to ACES server through ACES web site (http://aces.ee.olemiss.edu) by using the upload button, journal section. Only pdf files are accepted for submission. The file size should not be larger than 5MB, otherwise permission from the Editor-in-Chief should be obtained first. The Editor-in-Chief will acknowledge the electronic submission after the upload process is successfully completed.

COPYRIGHTS AND RELEASES

Each primary author must sign a copyright form and obtain a release from his/her organization vesting the copyright with ACES. Copyright forms are available at ACES, web site (http://aces.ee.olemiss.edu). To shorten the review process time, the executed copyright form should be forwarded to the Editor-in-Chief immediately after the completion of the upload (electronic submission) process. Both the author and his/her organization are allowed to use the copyrighted material freely for their own private purposes.

Permission is granted to quote short passages and reproduce figures and tables from and ACES Journal issue provided the source is cited. Copies of ACES Journal articles may be made in accordance with usage permitted by Sections 107 or 108 of the U.S. Copyright Law. This consent does not extend to other kinds of copying, such as for general distribution, for advertising or promotional purposes, for creating new collective works, or for resale. The reproduction of multiple copies and the use of articles or extracts for commercial purposes require the consent of the author and specific permission from ACES. Institutional members are allowed to copy any ACES Journal issue for their internal distribution only.

PUBLICATION CHARGES

ACES members are allowed 12 printed pages per paper without charge; non-members are allowed 8 printed pages per paper without charge. Mandatory page charges of $75 a page apply to all pages in excess of 12 for members or 8 for non-members. Voluntary page charges are requested for the free (12 or 8) pages, but are NOT mandatory or required for publication. A priority courtesy guideline, which favors members, applies to paper backlogs. Authors are entitled to 15 free reprints of their articles and must request these from the Managing Editor. Additional reprints are available to authors, and reprints available to non-authors, for a nominal fee.

ACES Journal is abstracted in INSPEC, in Engineering Index, and in DTIC.