ISTANBUL UNIVERSITY –
JOURNAL OF ELECTRICAL & ELECTRONICS ENGINEERING

YEAR : 2003 (955-959)
VOLUME : 3
NUMBER : 2

DESIGN OF LOW-PASS LADDER NETWORKS WITH MIXED LUMPED AND DISTRIBUTED ELEMENTS BY MEANS OF ARTIFICIAL NEURAL NETWORKS

Metin ŞENGÜL[1] Atilla ÖZMEN[2] Melek YILMAZ[3]

Kadir Has University, Engineering Faculty, 34230 Cibali, Fatih, İstanbul
Tel: +90 212 534 10 34

[1] E-mail: msengul@khas.edu.tr [2]E-mail: aozmen@khas.edu.tr

[3]E-mail: myilmaz@khas.edu.tr

ABSTRACT

In this paper, calculation of parameters of low-pass ladder networks with mixed lumped and distributed elements by means of artificial neural networks is given. The results of ANN are compared with the values that are desired. It has been observed that the calculated and the desired values are extremely close to each other. So this algorith can be used to obtain the parameters that will be used to synthesize such circuits.

Keywords: *Lumped and distributed elements, artificial neural networks, back propagation*

I. INTRODUCTION

One of the fundamental problems in the design and development of communication systems is to match a given device to the system via coupling circuits so as to achieve optimum performance over the broadest possible frequency band. This problem inherently involves the design of an equalizer network to match the given complex impedances, and usually referred as *impedance matching* or *equalization*.

It has been appreciated for quite long time that networks containing both lumped and distributed elements offer many agvantageous associated with the present day emphasis on integrated circuits and more suitable especially for microwave filter applications, than those containing only lumped or distributed elements. The design of mixed lumped-distributed structures has been studied by several authors, by using as well as practical design techniques [1] and also by multivariable network synthesis techniques [1]. The problem has gained renew interest because of the possible application of the

Received Date : 05*.01.2003*
Accepted Date: 06.06.2003

mixed element structures as the reference networks for the design of multidimensional Wave Digital Filters [1].

In the second part of this paper, characterization of two-dimensional lossless two-ports by scattering approach will be given. The third part belongs to artificial neural networks. In the succeeding part, the calculation of the 5th degrees circuit (3 lumped and 2 distributed elements) parameters that will be used to synthesize the circuit and the obtained results can be found.

2. CHARACTERIZATION OF TWO-DIMENSIONAL LOSSLESS TWO-PORTS BY SCATTERING APPROACH

In the theory of one-dimensional lossless two-ports, the so called scattering matrix and the corresponding scattering transfer matrix have turned out to be the most appropriate tools for establishing the synthesis theory. Of particular interest is the use of the Belevithc canonic forms of these matrices [1]. It is already known that, in the study of two-dimensional lossless two-ports, the scattering approach can be adopted as basis for the investigations.

Fettweis has shown that, the scattering matrix of a multidimensional reactance two-port with possible unit elements, as well as the corresponding scattering transfer matrix can be expressed in terms of canonic polynomials f, g and h [1]. In arriving at this description, the derivation follows essentially the same lines as in the one dimensional case, with the introduction of a particular type of multidimensional Hurwitz polynomials, namely Scattering Hurwitz or Principle Hurwitz polynomials, which plat yhe same role in multidimensional characterization as the classical Hurwitz polynomials so in one-dimensional case [1].

Before we consider the description of lossless two-ports by canonic polynomials, let us give first some basic definitions and the notation which will be used for the two-variable polynomials in the succeeding sections.

A polynomial g in two variables p,λ will be denoted by g(p,λ) or simply by g. g can be written as a polynomial in p and λ as

$$g(p,\lambda)=\sum_{k=1}^{n_p}\sum_{l=1}^{n_\lambda}g_{kl}p^k\lambda^l \qquad (1)$$

where n_p and n_λ are the partial degrees of g in the variables p and λ respectively.

Definition 1: A two-variable polynomial g(p,λ) is said to be of degree m, if $m=\max_{g_{kl}\neq 0}(k+l)$. In terms of the partial degrees n_p and n_λ it is defines as $m=n_p+n_\lambda$.

Arranging with respect to one of the variables with coefficient polynomials in the other variable, we obtain the recursive canonic form,

$$g(p,\lambda)=\sum_{k=1}^{n_p}G(\lambda)p^k=\sum_{l=1}^{n_\lambda}\bar{G}(p)\lambda^k \qquad (2)$$

Another form that we adopt for the representation of the two-variable polynomials is in matrix notation,

$$g(p,\lambda)=p^T\Lambda_g\lambda^T \qquad (3)$$

where $p^T=\lfloor 1\ p\ p^2\dots p^{n_p}\rfloor$ and $\lambda^T=\lfloor 1\ \lambda\ \lambda^2\dots\lambda^{n_\lambda}\rfloor$ and $n_p x n_\lambda$ coefficient matrix Λ_g is defined by

$$\Lambda_g=\begin{bmatrix} g_{00} & g_{01} & \cdots & g_{0n_\lambda}\\ g_{10} & g_{11} & \cdots & g_{1n_\lambda}\\ \vdots & \cdots & & \vdots\\ g_{n_p0} & g_{n_p1} & \cdots & g_{n_pn_\lambda}\end{bmatrix} \qquad (4)$$

The paraconjugate of the two-variable polynomials is defined as $g_*=g(-p,-\lambda)$.

Definition 2: A two-variable polynomial g(p,λ) is called as a mth-degree polynomial with no missing terms if it is expressible as

$$g(p,\lambda)=\sum_{k=0}^{m}\left(\sum_{l=0}^{m-k}g_{kl}\lambda^{l}\right)p^{k} \qquad (5)$$

where g_{ij} are nonzero real constants. In matrix notation, this corresponds to the nonzro upper-triangular coefficient matrix form.

$$\Lambda_g=\begin{bmatrix} g_{00} & g_{01} & \cdots & g_{0n_\lambda} \\ g_{10} & g_{11} & \cdots & 0 \\ \vdots & \cdots & & 0 \\ g_{n_p0} & 0 & \cdots & 0 \end{bmatrix} \qquad (6)$$

Now, let us consider two-variable characterization of lossless two-ports with mixed lumped-distributed elements and state the properties of canonic polynomials.

Consider the generic form of a lossless two-port formed with cascade connections of lumped and distributed two-ports as shown in Fig. 1.

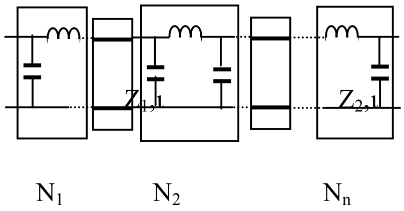


Figure 1 Generic form of cascaded lumped and distributed two-ports

The scattering matrix describing the mixed element two-port can be expressed in the Belevitch canonical form as [1],

$$S(p,\lambda)=\frac{1}{g(p,\lambda)}\begin{pmatrix} h(p,\lambda) & \sigma f(-p,-\lambda) \\ f(p,\lambda) & -\sigma h(-p,-\lambda) \end{pmatrix} \qquad (7)$$

where

- $f(p,\lambda)$ is a monic real polynomial that consist of transmission zeros,
- $h(p,\lambda)$ ve $g(p,\lambda)$ are real polynomials in the complex variables $p=\sigma+jw$ and $\lambda=\varepsilon+j\Omega$ ($\lambda=\tanh(p\tau)$, τ being the delay length of unit elements),
- $g(p,\lambda)$ is a Scattering Hurwitz polynomial,
- σ is a unimodular constant; $|\sigma|=1$,
- Losslessness of the mixed element two-port requires that

$$g(p,\lambda)g(-p,-\lambda)=h(p,\lambda)h(-p,-\lambda)+f(p,\lambda)f(-p,-\lambda)$$

.

From the physical implementation point of view, one practical circuit configuration is that of simple low-pass ladder sections connected with unit elements (LPLUE) as shown in Fig. 2.

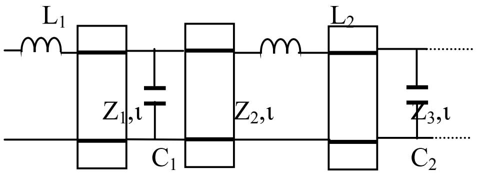


Figure 2 Low-pass ladder with unit element

In the Table 1, the explicit formulas for 5th order LPLUE are given.

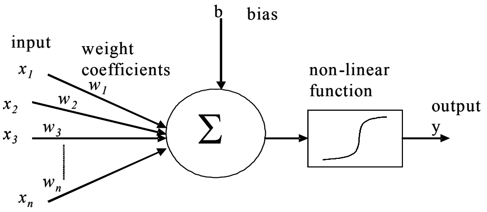
n	Coefficient Relations
5	[h01,h02,h10,h20,h30] independent coefficients $g_{01}=(2+2g_{02}+h_{01}^2)^{1/2}$ $g_{11}=g_{10}g_{01}-h_{10}h_{01}$ $g_{02}=(1+h_{02}^2)^{1/2}$ $h_{11}=h_{02}\beta/\alpha+h_{20}\alpha/\beta$ $g_{10}=(h_{10}^2+2g_{20})^{1/2}$ $g_{12}=(g_{11}g_{02}-h_{11}h_{02})/\alpha$ $g_{30}=\lvert h_{30}\rvert$, $\mu=h_{30}/g_{30}$ $\alpha=g_{01}-\mu h_{01}$ $h_{12}=\mu g_{12}$, $h_{21}=\mu g_{21}$ $\beta=g_{10}-\mu h_{10}$ $g_{20}=(h_{20}^2+2g_{10}g_{30}-2h_{10}h_{30})^{1/2}$ $g_{21}=(g_{11}g_{20}-h_{11}h_{20}-g_{01}g_{30}+h_{01}h_{30})/\beta$

3. NEURAL NETWORKS

Artificial neural networks (ANN) which consist of simplified neurons connected to each other are the models of nervous system. Although each neuron has a simple function alone, they can be used to solve complex problems, when they are used together.

Artificial neural networks are adaptive systems which have learning capabilities. ANNs adapt and organize themselves to the changing conditions, improve a function and make the calculation by learning. ANNs can produce the correct response even though missing or corrupted input is given to them. They are more suitable for the daily life problems because of their nonlinear characteristics [2].

In Figure 3, a neuron can be seen which consists of a summing junction and a non-linear activation function. Here, x_1, x_2, …,x_n are inputs; w_1, w_2,…, w_n are synaptic weight coefficients and y is output.



$$u=w_0x_0+w_1x_{1+}w_2x_2+...+w_nx_n$$

$$y=f(u+b)$$

Figure 3 Neuron model

A neural network model can be seen in Figure 4. Each neuron has many inputs and only one output, and this output is the input for the other neurons, so system is formed in parallelly.

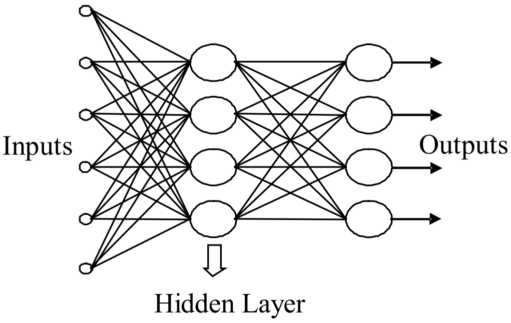


Figure 4 ANN with one hidden layer

ANNs can be used in signal processing, image processing, pattern recognition, medical, military systems, finance systems, artificial intelligence and power systems.

3.1. Learning in Artificial Neural Networks

Learning in artificial neural networks is based on the calculation of the synaptic weight coefficients suitable for the problem. Learning rule is an equation set by which all or some of the synaptic weight coefficients change so as to modify the response of each neuron in time. By this way ANN can adapt itself to get the desired response.

ANNs are learnt by example data instead of programming. Learning process can be divided into two groups; supervised and unsupervised learning.

In supervised learning, both the input and the response are given to the system. For each input, obtained response and desired response are compared. To get the minimum difference, synaptic weight coefficients are changed. When an acceptable error is obtained, learning process is stopped and then these synaptic weight coefficients can be used with the data that are not used in learning process.

3.2. Back Propagation Algorithm

In this paper, back propagation algorithm is used as the learning algorithm, which has emerged as the most widely used and successful algorithm for the design of multiplayer feedforward networks [3].

In learning process, first of all, an error is obtained by subtracting the result from the desired value. Then the error is squared. In this algorithm, it is desired to realize a learning process with an error whose square is minimum.

$$\varepsilon^2 = (t-y)^2 \tag{8}$$

$$\varepsilon = \varepsilon(y(u(w))) \tag{9}$$

Then delta values are calculated at the output nodes.

$$\nabla_w \varepsilon^2 \equiv \frac{\partial \varepsilon^2}{\partial w} \tag{10}$$

And by using back propagation, all the values at the output nodes are calculated.

$$\nabla_w \varepsilon^2 = \frac{\partial \varepsilon^2}{\partial y}\cdot\frac{\partial y}{\partial u}\cdot\frac{\partial u}{\partial w}$$

$$= -2\,\varepsilon\, sgm'(u)x \qquad (11)$$

$$= 2\,\delta\, x$$

$$\delta \equiv -\varepsilon\, sgm'(u) \qquad (12)$$

Finally, the components of gradient are calculated and desired synaptic weight coefficients are obtained.

$$\Delta w = -2\,\mu\,\delta\, x \qquad (13)$$

$$w_{new} = w_{old} + \Delta w \qquad (14)$$

4. EXAMPLE AND CONCLUSION

In this section, we solve the classical double matching problem. Here, an LPLUE network of degree five, (np=3, nλ=3) will be constructed over the normalized frequency band of 0≤w≤1.

In the course of the learning process, the coefficients {h00,h01,h02,h10,h20,h30,g00,g01,g02,g10,g20, g30} are chosen as the input parameters, the other coefficients are derived by means of designed neural network.

We obtained the following coefficient matrices that completely describe the scattering parameters of the matching network under consideration. Also you can see the matrices that are derived via the explicit expressions given in Table 1.

$$Ah=\begin{bmatrix} 0 & 1.7310 & -1.6281 \\ 0.1042 & -0.3420 & -1.6108 \\ 0.1827 & -3.5316 & 0 \\ -0.9960 & 0 & 0 \end{bmatrix} Ag=\begin{bmatrix} 1 & 2.9694 & 1.9107 \\ 2.0396 & 5.8760 & 1.6108 \\ 2.0746 & 3.5316 & 0 \\ 0.9960 & 0 & 0 \end{bmatrix}$$

$$Ah=\begin{bmatrix} 0 & 1.7310 & -1.6281 \\ 0.1042 & -5.6353 & -5.4744 \\ 0.1827 & -2.3089 & 0 \\ -0.9960 & 0 & 0 \end{bmatrix} Ag=\begin{bmatrix} 1 & 2.9694 & 1.9107 \\ 2.0396 & 6.7664 & 4.3960 \\ 2.0746 & 6.3454 & 0 \\ 0.9960 & 0 & 0 \end{bmatrix}$$

h and g matrices derived by ANN and explicit expressions, respectivelly.

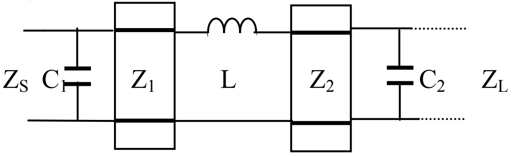


Fig 5. Double matching example with lumped-distributed elements [L=2.1439, C_1=0.8825, C_2=1.0528, Z_1=1.0356, Z_2=3.6650, ι=0.2191], [Z_S=R_S+jwL_S, Z_L=(R_L//(1/jwC_L)+jwL_L)]

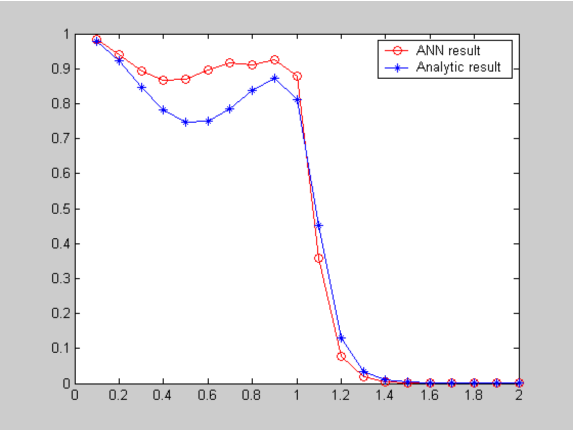


Fig 6. Gain performance of double matchingexample

5. REFERENCES

[1] Aksen, A., “Design of Lossloss Two-ports with Mixed Lumped and Distributed Elements for Broadband Matching”, Doktora tezi, Ruhr Üniversitesi, Bochum, 1994

[2] HAYKIN, S., “Neural Networks” Macmillan College Publishing Company Inc., 1994, ISBN: 0-02-352761-7

[3] Şengül.M, Özmen.A, Yılmaz.M, “Toplu ve Dağınık Elemanlı Alçak-Geçiren Merdiven Tipi Devre Parametrelerinin Yapay Sinir Ağları Kullanılarak Hesaplanması”, Eleco 2002, Bursa