



2012 Special Issue

Orthogonal least squares based complex-valued functional link network

Md. Faijul Amin^{a,*}, Ramasamy Savitha^b, Muhammad Ilias Amin^c, Kazuyuki Murase^a^a Department of System Design Engineering, University of Fukui, Fukui, 910-8507, Japan^b School of Electrical and Electronics Engineering, Nanyang Technological University, 639798, Singapore^c Department of Computer Science and Engineering, United International University, Dhaka, 1209, Bangladesh

ARTICLE INFO

Keywords:

Functional link network
Complex-valued neural network
Orthogonal least squares
Function approximation
Multivariate polynomial

ABSTRACT

Functional link networks are single-layered neural networks that impose nonlinearity in the input layer using nonlinear functions of the original input variables. In this paper, we present a fully complex-valued functional link network (CFLN) with multivariate polynomials as the nonlinear functions. Unlike multilayer neural networks, the CFLN is free from local minima problem, and it offers very fast learning of parameters because of its linear structure. Polynomial based CFLN does not require an activation function which is a major concern in the complex-valued neural networks. However, it is important to select a smaller subset of polynomial terms (monomials) for faster and better performance since the number of all possible monomials may be quite large. Here, we use the orthogonal least squares (OLS) method in a constructive fashion (starting from lower degree to higher) for the selection of a parsimonious subset of monomials. It is argued here that computing CFLN in purely complex domain is advantageous than in double-dimensional real domain, in terms of number of connection parameters, faster design, and possibly generalization performance. Simulation results on a function approximation, wind prediction with real-world data, and a nonlinear channel equalization problem exhibit that the OLS based CFLN yields very simple structure having favorable performance.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Complex-valued data arise in various applications, such as array and radar signal processing, magnetic resonance imaging, communication systems, signal representation in complex baseband, and processing data in the frequency domain (Hirose, 2006). Apart from this, some real-valued two dimensional data can have better representation as a complex vector field, for example, wind speed and direction (Mandic, Javidi, Goh, Kuh, & Aihara, 2009) and tree representation of hand in the hand gesture recognition system (Ghani, Amin, & Murase, 2011). Since artificial neural networks (ANNs) are well established efficient models for processing real-valued data, several studies have extended the ANNs to the complex domain with a view to utilizing their strength of nonlinear processing abilities (Benvenuto & Piazza, 1992; Georgiou & Koutsougeras, 1992; Leung & Haykin, 1991). It is, however, realized that such extensions are not trivial. A major difficulty involves selecting suitable nonlinear activation functions (Kim & Adali, 2003; Nitta, 1997) which enables the ANNs to capture nonlinear relationship between

input and output. In the real domain, there are a bunch of activation functions having two desirable properties, *boundedness* and *differentiability*. But in the complex domain, one has to adopt the trade-off between boundedness and analyticity due to Liouville's theorem: a bounded entire function is a constant in the complex domain.

An *ad hoc* approach to deal with complex-valued data is to consider the real and imaginary parts separately viewing a mapping $f : \mathbb{C}^m \rightarrow \mathbb{C}^n$ alternatively by the mapping $g : \mathbb{R}^{2m} \rightarrow \mathbb{R}^{2n}$, and then to solve the problem in real domain (Patra, Meher, & Chakraborty, 2009; Patra, Pal, Baliarsingh, & Panda, 1999). However, such an approach does not exploit the advantage of complex algebra which is the distinctive feature of complex-valued neural networks (CVNNs). Consequently, this real-valued perspective performs poorly in terms of efficient architecture, convergence and generalization ability (Nitta, 1997). A better approach is to use well defined complex algebra and split-type activation functions (Hirose, 2006; Nitta, 1997). The split-type activation functions are the popular sigmoid functions (logistic or hyperbolic tangent) applied to the real and imaginary parts (or sigmoid to the magnitude and identity function to the phase in polar coordinates) separately. However, there is still a problem that such functions are inefficient in complex-valued nonlinear mapping and are unable to provide true gradient in the error backpropagation learning process (Kim & Adali,

* Corresponding author. Tel.: +81 0 90 9440 7120; fax: +81 0 776 27 8420.

E-mail addresses: amin@synapse.his.u-fukui.ac.jp (M.F. Amin),
savi0001@e.ntu.edu.sg (R. Savitha), ilias.amin@gmail.com (M.I. Amin),
murase@synapse.his.u-fukui.ac.jp (K. Murase).

2003). This shortcoming is mitigated by adopting elementary transcendental functions at the cost of singularities of various kinds. The functions, however, have widely varying response in the vicinity of discontinuities which may cause a detrimental effect during the learning process. It seems that there is no conclusive remark as to how one should choose activation function for a given problem.

Along with the complex-valued multilayer perceptron (CMLP) (Kim & Adali, 2003; You & Hong, 1998), there have been a growing interests in developing complex-valued radial basis function (CRBF) networks. One of the early works by Chen, Grant, McLaughlin, and Mulgrew (1993) extends the real RBF by using complex centers and weights. A potential application of CRBF networks for solving communication channel equalization with a stochastic gradient training algorithm is demonstrated in Cha and Kassam (1995). In order to design and ensure a parsimonious RBF network by growing and pruning strategies, Jianping, Sundararajan, and Saratchandran (2002) have proposed a sequential learning algorithm, referred to as complex minimal resource allocation network (CMRAN). Even though the aforementioned approaches employ complex-valued centers, the response of hidden units remains real as the hidden units use Gaussian RBFs with Euclidean norm. Hence, they cannot approximate input–output mapping efficiently, especially for the phase values. This has been overcome with the development of a fully complex-valued RBF (FC-RBF) network with a fully complex-valued activation function (Savitha, Suresh, & Sundararajan, 2009). It is shown that the FC-RBF outperforms the conventional CRBF networks in its approximation ability. The authors have also proposed an efficient learning scheme for the FC-RBF that use a self-regulatory system (Savitha, Suresh, & Sundararajan, 2010). The activation function used in FC-RBF, however, is one of the elementary transcendental function having singularities.

Besides the issue of activation function in complex domain, the complex-valued multilayer structures (CMLPs and CRBF networks) share similar problems with the networks in real domain. The Multilayer structures are computationally intensive because of nonlinear projections by the hidden layers (Patra et al., 1999). Most importantly, their training is difficult due to several problems, including local minima trapping, saturation of activation functions, slow convergence, initial weight dependence, and overfitting due to structural complexity (Sierra, Macias, & Corbacho, 2001).

The problems stated above can be avoided by removing the hidden layers, but without giving up the nonlinearity. Instead, nonlinearity can be imposed in the input layer by functional expansion of input variables. The resulting model has a flat structure, i.e., single layer network. This alternative approach is known as functional link networks (FLNs) in the literature (e.g., Pao, 1989) and advocated for alleviating the previously mentioned problems of multilayer structures. However, the price to be paid for is the proper choice of functional expansion, for example, polynomials, trigonometric functions, or orthogonal basis functions.

In the real domain, many works on the FLNs can be found encompassing pattern recognition, control applications, and communication channel equalization (Dehuri & Cho, 2010b; Pao & Phillips, 1995; Patra & Pal, 1995). Only few attempts have been taken toward the application of FLNs for processing complex-valued data. In Patra et al. (1999), despite the data being complex, the authors have considered the problem in the real domain separating the real and imaginary parts. This real-valued perspective does not exploit the well defined complex algebraic rules, and hence, cannot provide efficient modeling.

In this study, we present a complex-valued functional link network (CFLN) for solving complex-valued function approximation problems. In order to capture nonlinear relationship between the

input and output, multivariate polynomials are considered. In contrast to other possibilities like elementary transcendental functions, polynomials are easy to compute and include higher order cross product terms. Moreover, polynomials are equally well behaving in the real and complex domain, while most of the transcendental functions have singularities in the complex domain (Kim & Adali, 2003). Since the number of total terms (or *monomials*) in the multivariate polynomial grows exponentially with the degree and number of input variables, only the relevant monomials are selected in the CFLN by the orthogonal least squares (OLS) method. The Polynomial degree is increased in an incremental way until a negligible improvement is seen from the increment. In this regard, it is worth mentioning that computing with polynomials are also prevalent in other CVNNs because the nonlinear functions are often evaluated by the polynomial expansions. Thus, using polynomials in the input layer does not involve additional computational cost than the other CVNNs.

To the best of our knowledge, our study is the first of its kind in the FLNs that considers complex-valued data directly, without separating the real and imaginary parts. The proposed CFLN offers a number of advantages over conventional CVNNs. First, it does not require activation functions which has been a major concern in the complex domain till now. Second, the cost function, i.e., mean squared error (MSE), has a single minimum as it is quadratic in parameters. Therefore, fast learning algorithms such as OLS or recursive least squares (Sayed, 2008) can be used to learn the weight parameters in the CFLN. In contrast, learning in the multilayer architectures often becomes difficult due to the local minima problem and slow convergence of error backpropagation. Third, selecting a near optimal set of monomials by the OLS method yields very simple CFLNs with favorable performance.

The remaining sections of the paper are structured as follows. Section 2 provides a brief overview of FLNs in a general framework. A detailed description of CFLN design by the OLS method is presented in Section 3. Experimental results comprising a complex-valued function approximation, a channel equalization, and a real-world wind prediction problem are provided in Section 4. Finally, conclusion is given in Section 5.

2. Brief overview of FLNs

The FLNs are single-layered networks without hidden layers, where the input layer is formed by some predefined functions of input variables in addition to the original variables. The resulting network is flat as shown in Fig. 1. Let the original variables be $\mathbf{x} = (x_1, x_2, \dots, x_N)$. Then the input layer is constructed as $(x_1, x_2, \dots, x_N, \phi_1(\mathbf{x}), \dots, \phi_M(\mathbf{x}))$. The output units are simply linear combinations of the enhanced input units; each output can be written as

$$y_k = \sum_{i=1}^N \alpha_{ki} x_i + \sum_{j=1}^M \beta_{kj} \phi_j(\mathbf{x}) \quad 1 \leq k \leq K \quad (1)$$

where the functions $\phi_j(\mathbf{x})$, $1 \leq j \leq M$, are linearly independent, for example, polynomials, trigonometric, or hyperbolic functions of input variables. The symbol K denotes the total number of output units.

In the multilayer networks like MLPs, successive layers carry out a sequence of mappings in order to enhance original input representation (Nilsson, 1965) with the help of a supervised learning method. The learning, however, suffers from various limitations, such as slow convergence, local minima trapping, initial weight dependence, and saturation of activation functions making the learning process difficult. The viewpoint of enhancing input representation in the FLNs is same as in MLPs, but the enhancement is carried out right from the start in the input layer, in

Download English Version:

<https://daneshyari.com/en/article/404341>

Download Persian Version:

<https://daneshyari.com/article/404341>

[Daneshyari.com](https://daneshyari.com)