Available online at www.sciencedirect.com
ScienceDirect
Journal of the Franklin Institute 348 (2011) 415–425
Journal of The Franklin Institute
www.elsevier.com/locate/jfranklin

Short communication

Genetically optimized Hybrid Fuzzy Set-based Polynomial Neural Networks

Sung-Kwun Oh[a], Witold Pedrycz[b,c,*], Seok-Beom Roh[a]

[a]*Department of Electrical Engineering, The University of Suwon, San 2-2 Wau-ri, Bongdam-eup, Hwaseong-si, Gyeonggi-do 445-743, South Korea*
[b]*Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada T6G 2G6*
[c]*Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland*

Received 12 February 2005; received in revised form 14 September 2010; accepted 23 November 2010
Available online 1 December 2010

Abstract

We investigate a new category of fuzzy-neural networks such as Hybrid Fuzzy Set-based Polynomial Neural Networks (HFSPNN). These networks consist of a genetically optimized multi-layer with two kinds of heterogeneous neurons such as fuzzy set-based polynomial neurons (FSPNs) and polynomial neurons (PNs). We have developed a comprehensive design methodology that helps determine the optimal structure of networks dynamically. The augmented genetically optimized HFSPNN (referred to as gHFSPNN) results in a structurally optimized structure and comes with a higher level of flexibility in comparison to the one we encounter in the conventional HFPNN. The GA-based design procedure being applied at each layer of gHFSPNN leads to the selection of preferred nodes (FSPNs or PNs) available within the HFSPNN. In the sequel, the structural optimization is realized via GAs, whereas the ensuing detailed parametric optimization is carried out in the setting of a standard least square method-based learning. The performance of the gHFSPNN is demonstrated through intensive experimentation where we use a number of modeling benchmarks—synthetic and experimental datasets are already being used in fuzzy or neurofuzzy modeling.

1. Introduction

A lot of researchers involved in system modeling have been interested in the multitude of challenging and usually conflicting modeling objectives such as compactness, approximation

*Corresponding author.
E-mail address: pedrycz@ee.ualberta.ca (W. Pedrycz).

0016-0032/$32.00 © 2010 The Franklin Institute. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.jfranklin.2010.11.005

abilities, and generalization aspects of the proposed constructs. It is a common approach to experiment with various forms of neural networks and fuzzy systems in the design of models of a nonlinear system characterized by sound predictive abilities as good approximation characteristics. In particular, when dealing with high-order nonlinear and multivariable equations of the proposed models, we typically require a vast amount of data for estimating all its parameters. The Group Method of Data Handling (GMDH) [1] being introduced by Ivakhnenko is one of the approaches that help alleviate the problem. Having said that, we should be aware that GMDH comes with some drawbacks. First, it tends to generate quite complex polynomials even for relatively simple systems. Second, owing to its limited generic structure, GMDH also tends to produce an overly complex network (model) when it comes to highly nonlinear systems. In alleviating the problems of the GMDH algorithms, Polynomial Neural Networks (PNN)[2–4] were introduced as a new class of networks. The synergy of neural networks [5,6] and fuzzy systems [7] (giving rise to neurofuzzy systems) has been recognized as a powerful alternative approach to the development of fuzzy systems. We have investigated a new category of neurofuzzy networks, Fuzzy Set-based Polynomial Neural Networks (FSPNN) and developed Hybrid Fuzzy Set-based Polynomial Neural Networks (HFSPNN) composed of a multi-layer structure with two types of heterogeneous neurons, namely fuzzy set-based polynomial neurons (FPSNs) and polynomial neurons (PNs). Although the HFSPNNs exhibit flexible architectures whose potential can be fully utilized through a systematic design, it is difficult to obtain a structurally and parametrically optimized network because of the limited design of the individual nodes (viz. FSPNs and PNs) located in each layer of the network.

In this paper, we study a genetic optimization [8]-driven new neurofuzzy topology called genetically optimized Hybrid Fuzzy Set-based Polynomial Neural Networks (gHFSPNN) and discuss a comprehensive design methodology supporting their development. gHFSPNN is a network resulting from the combination of a fuzzy inference system and a PNN algorithm driven to genetic optimization. Each node of the first layer of gHFSPNN, i.e. a fuzzy polynomial neuron (FSPN), operates as a compact fuzzy inference system. The networks of the second and higher layers of the gHFSPNN come with a high level of flexibility as each node (processing element forming a PN). The determination of the optimal values of the parameters available within an individual PN and FSPN (viz. the number of input variables, the order of the polynomial, a collection of preferred nodes, and the number of membership functions (MFs)) leads to a structurally and parametrically optimized network.

2. Architecture of Hybrid Fuzzy Set-based Polynomial Neural Networks

2.1. Architecture of fuzzy set-based polynomial neurons (FSPN) based layer of gHFSPNN

The FSPN encapsulates a family of nonlinear "if–then" rules. When put together, FSPNs result in self-organizing Fuzzy Set-based Polynomial Neural Networks (FSPNN). The FSPN consists of two basic functional modules. The first one, denoted by F, is a collection of fuzzy sets (here denoted by $\{\boldsymbol{A}_k\}$ and $\{\boldsymbol{B}_k\}$) that form an interface between the input numeric variables and the processing part realized by the neuron. The second module (denoted here by P) refers to the function-based nonlinear (polynomial) processing that involves some input variables This nonlinear processing involves some input variables (x_i and x_j), which are capable of being the input variables (here, x_p and x_q), or entire system input variables.