Chapter 12
A Novel Neural Fuzzy Network Using a Hybrid Evolutionary Learning Algorithm

Cheng-Jian Lin
National Chin-Yi University of Technology, Taiwan, R. O. C.

Cheng-Hung Chen
National Chin-Yi University of Technology, Taiwan, R. O. C.

ABSTRACT

This chapter presents an evolutionary neural fuzzy network, designed using the functional-link-based neural fuzzy network (FLNFN) and a new evolutionary learning algorithm. This new evolutionary learning algorithm is based on a hybrid of cooperative particle swarm optimization and cultural algorithm. It is thus called cultural cooperative particle swarm optimization (CCPSO). The proposed CCPSO method, which uses cooperative behavior among multiple swarms, can increase the global search capacity using the belief space. Cooperative behavior involves a collection of multiple swarms that interact by exchanging information to solve a problem. The belief space is the information repository in which the individuals can store their experiences such that other individuals can learn from them indirectly. The proposed FLNFN model uses functional link neural networks as the consequent part of the fuzzy rules. This chapter uses orthogonal polynomials and linearly independent functions in a functional expansion of the functional link neural networks. The FLNFN model can generate the consequent part of a nonlinear combination of input variables. Finally, the proposed functional-link-based neural fuzzy network with cultural cooperative particle swarm optimization (FLNFN-CCPSO) is adopted in several predictive applications. Experimental results have demonstrated that the proposed CCPSO method performs well in predicting the time series problems.

INTRODUCTION

Prediction has been widely studied for many years as time series analysis (Box & Jenkins, 1970; Tong, 1990). Traditionally, prediction is based on a statistical model that is either linear or nonlinear (Li et al., 1990). Recently, several studies have adopted neural fuzzy networks to predict time series (Cowder, 1990; Kasabov & Song, 2002; Ling et al., 2003). Researchers have discussed that the network paradigm is a very useful model for predicting time series and especially for predicting nonlinear time series.

DOI: 10.4018/978-1-61520-757-2.ch012

Soft computing tools, including fuzzy sets, neural networks, and evolutionary algorithms, have been experimentally used to handle real-life ambiguous situations (Bhattacharya et al., 2007; Chu & Tsai, 2007; Karaboga & Basturk, 2008; Kim et al., 2006). Many existing soft computing techniques (Huang et al., 2006; Mitra et al., 2002; Yu, 2007; Zhang et al., 2000) are most widely applied to solve data mining problems especially for classification or prediction. Neural fuzzy networks (Angelov & Filev, 2004; Jang, 1993; Juang & Lin, 1998; Li & Lee, 2003; Lin, 2008; Lin & Chin, 2004; Lin & Lee, 1996; Sun et al., 2003; Takagi & Sugeno, 1985) have become a popular research topic. They bring the low-level learning and computational power of neural networks into fuzzy systems and bring the high-level human-like thinking and reasoning of fuzzy systems to neural networks. In the typical TSK-type neural fuzzy network (Angelov & Filev, 2004; Jang, 1993; Juang & Lin, 1998; Li & Lee, 2003; Lin, 2008; Sun et al., 2003; Takagi & Sugeno, 1985), which is a linear polynomial of input variables, the model output is approximated locally by the rule hyperplanes. However, the traditional TSK-type neural fuzzy network does not take full advantage of the mapping capabilities that may be offered by the consequent part. Introducing a nonlinear function, especially a neural structure, to the consequent part of the fuzzy rules has yielded the NARA (Takagi et al., 1992) and the CANFIS (Mizutani & Jang, 1995) models. These models (Mizutani & Jang, 1995; Takagi et al., 1992) use multilayer neural networks in the consequent part of the fuzzy rules. Although the interpretability of the model is reduced, the representational capability of the model is significantly improved. However, the multilayer neural network has such disadvantages as slower convergence and greater computational complexity. Therefore, we proposed the functional link neural fuzzy network (FLNFN), which uses the functional link neural network (FLNN) (Pao, 1989; Patra, 1999) in the consequent part of the fuzzy rules (Chen et al., 2007). The FLNN is a single layer neural structure that is capable of forming arbitrarily complex decision regions by generating nonlinear decision boundaries. Additionally, using functional expansion effectively increases the dimensionality of the input vector and the hyperplanes that are generated by the FLNN provide a good discrimination capability in input data space.

Training of the parameters is the main problem in designing a neural fuzzy system. Backpropagation (BP) training is commonly adopted to solve this problem. It is a powerful training technique that can be applied to networks with a forward structure. Since the steepest descent approach is used in BP training to minimize the error function, the algorithms may reach the local minima very quickly and never find the global solution.

The aforementioned disadvantages lead to suboptimal performance, even for a favorable neural fuzzy network topology. Therefore, technologies that can be used to train the system parameters and find the global solution while optimizing the overall structure, are required. Recently, many studies (Chatterjee et al., 2005; Huang, 2008; Juang, 2004; Wong et al., 2008) have received increasing attention mainly because they combine the neural fuzzy networks with the learning capabilities of swarm intelligence. Accordingly, a new optimization algorithm, called particle swarm optimization (PSO), appears to be better than the backpropagation algorithm. It is an evolutionary computation technique that was developed by Kennedy and Eberhart in 1995 (Eberhart & Kennedy, 1995; Kennedy & Eberhart, 1995). The underlying motivation for the development of PSO algorithm is the social behavior of animals, such as bird flocking, fish schooling and swarm theory. The major advantages of particle swarm optimization are as follows; 1) it has memory, so knowledge of good solutions is retained by all particles; 2) it has constructive cooperation between particles, particles in the swarm share information between them; 3) it has the fast global searching ability. PSO has been successfully applied to many