Optik 127 (2016) 1909–1917

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.de/ijleo

Multiobjective learning algorithm based on membrane systems for optimizing the parameters of extreme learning machine

Chuang Liu*, Dongling Chen, Fucai Wan

School of Information Engineering, Shenyang University, Liaoning 110044, China

ARTICLE INFO

Article history:
Received 18 May 2015
Accepted 17 November 2015

Keywords:
Membrane system
Multiobjective membrane algorithm
Membrane computing
Extreme learning machine

ABSTRACT

For adaptively learning the parameters of extreme learning machine (ELM), a novel learning algorithm is proposed on the basis of a multiobjective membrane algorithm. More specifically, first, a multiobjective mathematical model is established to learn the parameters of ELM, which is constructed by three objective functions. These objective functions include the root mean squared error, L_1 norm of output weights and the number of hidden nodes. Second, a series of the trade-off solutions with respect to the above-mentioned objective functions are found by the multiobjective membrane algorithm. Finally, a trade-off solution with the best generalization performance of ELM, which is chosen from the Pareto front obtained by the multiobjective algorithm, will become the final parameters for initializing the ELM network. The simulation experiments are run on the approximation of 'SinC' function, real-world regression problems and real-world classification problems. Experimental results indicate that the proposed framework is able to achieve good generalization performance in the most cases with many compact networks.

1. Introduction

Artificial neural networks (ANNs) are known as universal approximates and parallel distributed processing models which have been applied widely in many application fields such as classification, forecasting, optimization and system identification and so on [1,2]. When ANNs are employed to solve the real-world problems, they need to be firstly configured, including an appropriate network model, network topology and efficient training algorithm [3–6]. However, the good or bad configuration of ANNs will influence directly on their generalization performance. For example, unnecessary input weights and thresholds of ANNs not only increase the complexity of the model, but also increase the training time to unknown data. Traditional training algorithms such as backpropagation (BP) are usually slow and may get stuck in local minima. They make the network model to have a non-optimal performance, and the network model may suffer from the overfitting phenomenon (the learning model will have good results on training samples, but it will have bad results on test samples). These problems heavily limit the practical applications of ANNs.

A new and fast learning strategy is presented for a single hidden layer feedforward neural network, called extreme learning machine (ELM) [1,7]. Unlike traditional ANNs, ELM randomly chooses input weights and thresholds, and the Moore–Penrose (MP) method is employed to compute its output weights. However, in a given training cycle, the fixed parameters of ELM may not meet the optimum performance. Moreover, a small complex ELM leads to the limited capacity which can not provide good generalization performance. In contrast, a large one may have the good computation capacity, but it has some redundant network connections. At present, input weights and a structure of ELM are determined according to some priori knowledge. This way spends too much time on learning to adjust network configurations. And it is not conducive to the output stability of ELM. Recently, many researchers have proposed various effective learning algorithms which learn the parameters of the ELM in order to improve the generalization performance with a reasonable compact structure [8–12]. Therefore, the effective and automatic design of the ELM configuration has become a research topic.

In order to deal with the above dilemma more effectively, the use of evolutionary approaches to configure ELM has received increasing attention, because these approaches have a number of intuitive advantages without the priori knowledge in this domain [13,14]. Nowadays, evolutionary algorithms, including genetic algorithms (GA) [15,16], particle swarm optimization (PSO) [17], and differential evolution (DE) [18] and so on, are utilized to adaptively learn the configuring parameters of ELM. Therefore, to ensemble the self-learning ability of ELM and the adaptive ability of evolutionary algorithms are proposed, named as an evolutionary extreme learning machine. The evolutionary extreme learning machine may

* Corresponding author. Tel.: +86 24 62267995.
E-mail address: chuang.liu@mail.dlut.edu.cn (C. Liu).

http://dx.doi.org/10.1016/j.ijleo.2015.11.140
0030-4026/

automatically adjust the input weights, thresholds and the structure of ELM based on the sample data. It is worth pointing out that each solution, which includes the input weights, thresholds and the structure of ELM, is attained in the multi-dimensional solutions space by the evolutionary algorithm. In [9], a single-objective DE algorithm was employed to tune input weights and thresholds of ELM, called E-ELM. In [10], a new evolutionary ELM based on PSO was presented in prediction task. In [11], a combination of Integer Coded GA and PSO, coupled with the ELM had been used for gene selection and cancer classification. In [19,12], an optimally pruned ELM was proposed to improve the generalization performance of ELM by pruning the useless neurons. In [8], a structure-adjustable online learning neural network (SAO-ELM) based on the ELM with quicker learning speed and better generalization performance was proposed. These experiences are very useful for improving the generalization performance of ELM. However, previous studies to evolve the parameters of ELM consider only the generalization performance of the network. Moreover, L_1 norm of output weights and the number of hidden nodes have a direct impact on the overall performance of the network.

This paper aims to study whether the generalization performance of ELM can be further improved by employing a learning algorithm based on the multiobjective optimization algorithm. In other words, the learning algorithm is employed to adaptively learn the configuration of ELM. The detail of the algorithm is discussed as follows. Three objective functions related to the final generalization performance of ELM are established, which includes the root mean squared error, L_1 norm of output weights and the number of hidden nodes. And the three objective optimization problems need to be solved in order to find a suitable ELM network. On the basis of our previous work [20,21], a multiobjective optimization algorithm based on the membrane systems, called MOMC, is employed to find a series of trade-off solutions for the above-mentioned three objective optimization problem. Moreover, one of these trade-off solutions will be chosen according to the generalization performance of ELM. Subsequently, this learning algorithm in the simulation is applied to solve a regression problem of 'SinC' and the two real-world problems including the regression problems and the classification problems from the University of California at Irvine (UCI) repository [22]. The proposed framework, which is based on MOMC learning the parameters of ELM, is compared with the original ELM [7], E-ELM [9] and NSGAII-ELM (based on NSGAII [23] learning the parameters of ELM), respectively.

The remainder of this paper is organized as follows. Section 2 presents a brief review including the multiobjective optimization, ELM, and membrane systems. In Section 3, the hybrid framework of the multiobjective membrane algorithm and ELM are elaborated. Comprehensive study and experimental results are discussed in Section 4, and finally, Section 5 provides the concluding remarks of the study.

2. Preliminaries

2.1. Multiobjective optimization

Many real-world applications have existed with the competing multiple objectives simultaneously. This kind of problems, which include multiple conflicting objective functions, can be abstracted as multiobjective optimization problems (MOPs) [24]. The mathematical model of a MOP consists of the objective functions and constraint conditions. In general, the objective functions are a vector which consists of two or more than two conflict objective functions. Moreover the constraint conditions have two forms including the equality constraints and inequality constraints.

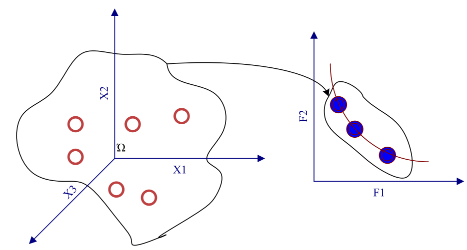

Fig. 1. The mapping between Pareto optimal solution set and Pareto front.

Without the loss of generality, the formal description of a MOP to be minimized is given in (1).

$$\begin{aligned} &F(X) = \min\{f_1(X), \ldots, f_m(X)\} \\ &s.t. \quad g_i(X) = 0, \quad i = 1, \ldots, q, \\ &h_j(X) \ge 0, \quad j = 1, \ldots, p, \\ &X_l \le X \le X_u, \quad X \in R^n, \quad F \in R^m \end{aligned} \tag{1}$$

where $X = (x_1, x_2, \ldots, x_n)$ is a decision vector including n dimensional decision variables. X_l is the lower boundary of the decision vector. X_u is the upper boundary of the decision vector. $F(X) = (f_1(X), \ldots, f_m(X))$ is an objective vector containing m objective functions. $g_i(X)$ denotes the ith equality constraints. $h_j(X)$ denotes the jth inequality constraints.

In order to understand the process of solving the multiobjective optimization problem, some definitions of multiobjective optimization are given as follows.

Definition 1 *(Pareto-dominate).* A decision vector $X = (x_1, x_2, \ldots, x_n)$ is said to dominate the other vector $V = (v_1, v_2, \ldots, v_n)$, if and only if both the statements below are satisfied.

$$\forall i \in 1, 2, \ldots, n : f_i(X) \le f_i(V) \bigwedge \exists j \in 1, 2, \ldots, n : f_j(X) < f_j(V) \tag{2}$$

Definition 2 *(Pareto-optimal set, X^*).* If no decision vector is dominated by the other decision vectors in the set Ω, these decision vectors constitute a Pareto-optimal set, denoted by X^* and shown on the left side of Fig. 1. The mathematical expression is described as $X^* = \{x \in \Omega, x' \in \Omega | \neg \exists x', x' \prec x\}$.

Definition 3 *(Pareto front, PF).* All decision vectors in X^* are mapped from the decision space to the objective spaces. Pareto front in the objective spaces is shown on the right side of Fig. 1. Its mathematical expression is described as $PF = f(x)|x \in X^*$.

2.2. Extreme learning machine

To overcome the slowly learning ability of traditional optimization techniques, Huang et al. [7] proposed ELM to train a single-hidden layer feedforward neural network (SLFNN) as shown in Fig. 2. Unlike the existing learning algorithm of SLFNN, the

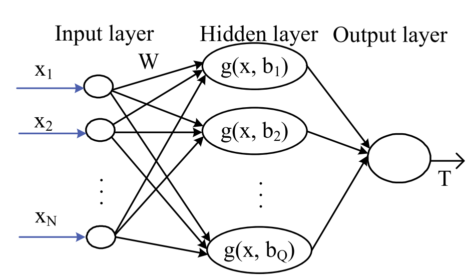

Fig. 2. A standard structure of ELM.