Engineering Applications of Artificial Intelligence 46 (2015) 258–268

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A comparative study of population-based optimization algorithms for downstream river flow forecasting by a hybrid neural network model

X.Y. Chen, K.W. Chau *, A.O. Busari

Department of Civil and Environmental Engineering, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China

ARTICLE INFO

Article history:
Received 10 February 2015
Received in revised form
18 September 2015
Accepted 18 September 2015
Available online 23 October 2015

Keywords:
Differential evolution
Artificial bee colony
Ant colony optimization
Downstream river flow forecasting
Hybrid neural network

ABSTRACT

Population-based optimization algorithms have been successfully applied to hydrological forecasting recently owing to their powerful ability of global optimization. This paper investigates three algorithms, i.e. differential evolution (DE), artificial bee colony (ABC) and ant colony optimization (ACO), to determine the optimal one for forecasting downstream river flow. A hybrid neural network (HNN) model, which incorporates fuzzy pattern-recognition and a continuity equation into the artificial neural network, is proposed to forecast downstream river flow based on upstream river flows and areal precipitation. The optimization algorithm is employed to determine the premise parameters of the HNN model. Daily data from the Altamaha River basin of Georgia is applied in the forecasting analysis. Discussions on the forecasting performances, convergence speed and stability of various algorithms are presented. For completeness' sake, particle swarm optimization (PSO) is included as a benchmark case for the comparison of forecasting performances. Results show that the DE algorithm attains the best performance in generalization and forecasting. The forecasting accuracy of the DE algorithm is comparable to that of the PSO, and yet presents weak superiority over the ABC and ACO. The Diebold–Mariano (DM) test indicates that each pair of algorithms has no difference under the null hypothesis of equal forecasting accuracy. The DE and ACO algorithms are both favorable for searching parameters of the HNN model, including the recession coefficient and initial storage. Further analysis reveals the drawback of slow convergence and time-consumption of the ABC algorithm. The three algorithms present stability and reliability with respect to their control parameters on the whole. It can be concluded that the DE and ACO algorithms are considerably more adaptive in optimizing the forecasting problem for the HNN model.

1. Introduction

River flow forecasting is a prerequisite for many water resource applications such as flood warning and reservoir design. The hydrological process of river flows is so complex that simple data-driven models are not able to describe its behavior. It is therefore worth investigating suitable models for the highly-nonlinear and seasonal river flows. The reasoning ability of fuzzy-based neural networks has led to an increasing interest within the hydrology community. So far a number of neuro-fuzzy computing techniques and fuzzy set theories have already been applied in hydrological modeling (Li and Chen, 2010; Tabari et al., 2012; Rath et al., 2013). Incorporating a fuzzy concept activation function into an artificial neural network (ANN) model is a good alternative to the latter since the former can be flexibly implemented. Qiu et al. (1998) introduced a fuzzy pattern-recognition activation function (from the input layer to the hidden layer) into an ANN model to forecast annual runoff. This function classified runoff into a number of categories during wet and dry seasons, and therefore reflected nonlinear and seasonal characteristics of the river system. Zhao and Chen (2008) developed a mixed forecasting model combining neural networks with fuzzy pattern-recognition, considering the fuzziness in the concept of similar basins. This practice, in which activation functions in the ANN structure are modified with characterization, is more easily executed and also addresses the fuzzy behavior of river flows. In this paper, this type of fuzzy pattern-recognition activation function is introduced into a hybrid neural network (HNN) model, representing uncertainties of the river flow problem.

* Corresponding author.
E-mail address: cekwchau@polyu.edu.hk (K.W. Chau).

http://dx.doi.org/10.1016/j.engappai.2015.09.010
0952-1976/

As an important issue in hydrological modeling, conceptual models have been developed for river flow forecasting over the past few decades. Rainfall-runoff models have extensively emerged in river flow forecasting by considering simplified forms of physical laws (Moore, 2007; Bhadra et al., 2010; Rezaeianzadeh et al., 2013). Another practical approach is to directly integrate physical equations into the ANN models. Yang et al. (1998) developed a hydrological modeling network (HYMN) model using the continuity equation and considering nodes in the hidden or output layers as storage reservoirs. This method has been successful applied to compute the monthly river flow at the Salford University station in the Irwell River basin using daily evaporation data and precipitation records from six upstream stations. Li and Gu (2003) further expanded the HYMN model to sediment yield forecasting. These investigations are rather limited, whereas they attain the possibility of directly integrating fundamental physical principles into an ANN structure. This paper contributes to assembling the fuzzy pattern-recognition concept and continuity equation into the ANN framework, with an explicit emphasis on the seasonal and non-stationary features of the river flows.

The proposed forecasting model is highly nonlinear and varying with time. Traditional optimization algorithms, e.g. the Gauss–Newton algorithm and gradient descent method are not applicable for this non-differentiable and multi-dimensional problem. Recently, population-based optimization algorithms have attracted the interest of many researchers (Blum and Roli, 2003; Chiong et al., 2012). They are characterized by a population consisting of possible solutions to the problem, which are modified by applying different types of operators and thus moving towards a near-optimal solution area. These algorithms are very powerful in finding a global optimum since they simultaneously search in many directions by using a population of possible solutions. Generally, there are two categories of population-based optimization algorithms: evolutionary algorithms and swarm intelligence algorithms (Blum et al., 2012). Typical methods of evolutionary algorithms are the genetic algorithm (GA) and differential evolution (DE), which attempt to simulate natural evolution. The DE algorithm was proposed by Storn and Price (1995) and has been applied extensively in hydrological modeling (e.g., Babu and Angira, 2003; Vasan and Simonovic, 2010; Liu and Pender, 2013). It shows better performances than the GA in terms of convergence characteristics and computational efficiency (Wang et al., 2008; Li et al., 2013; Song et al., 2014). In view of its ability to handle optimization problems that are non-differentiable, nonlinear, non-continuous and varying with time (Rocca et al., 2011), it is adopted in this paper for the comparative study.

The second category, swarm intelligence-based algorithms are inspired by the collective behavior of animal societies, including particle swarm optimization (PSO), artificial bee colony (ABC) and ant colony optimization (ACO). This paper focuses on ABC and ACO as stochastic global optimization algorithms. The ABC algorithm was introduced and popularized by Karaboga (2005) to solve numerical optimization problems. It has predictive capability comparable to the GA, PSO and DE algorithms on numerical test functions (Karaboga and Akay, 2009). Hybrid models that combine ABC algorithms with ANNs have been developed recently (Karaboga et al., 2007; Kisi et al., 2012). The ABC algorithm has been found to apply in many fields, e.g. HVAC systems (Zhang et al., 2013), reservoir optimum problems (Hossain and El-shafie, 2013) and protein structure optimization (Li et al., 2015). Another swarm-based optimization method ACO was derived from the food searching behavior of ants (Dorigo et al., 1996). Similar to the ABC algorithm, it is a meta-heuristic technique available to solve non-linear optimization problems with high dimensionality and inequality constraints. Coupling ACO algorithms with feed-forward neural network training has proven to be successful (Li and Chung, 2005; Shelokar et al., 2007; Socha and Blum, 2007). The potential to apply ACO to the field of river flow forecasting is clear, e.g., see its application in water resource problems (Maier et al., 2003; Jalali et al., 2006; Kumar and Reddy, 2006).

Optimization of neural networks has always been an open research. It is imperative to solve the disadvantages of traditional learning algorithms, such as poor generalization, slow convergence speed and easily plunging into local optima. The main objective of this paper is therefore to incorporate population-based optimization algorithms (i.e. DE, ABC and ACO) into a HNN model and compare their optimization ability, stability and reliability, and thereby determine the most adaptive optimization algorithm for the river flow forecasting problem.

The rest of this paper is structured in the following manner. A description of the HNN model for downstream river flow forecasting is presented in Section 2. In Section 3, a brief review of the three population-based optimization algorithms is provided. Section 4 introduces the case study site, while Section 5 summaries the computational results and comparisons of different optimization algorithms. Finally, the conclusion is reported in Section 6.

2. Hybrid neural network (HNN) model

The ANN is an efficient data-driven model for real-time forecasting. For a typical three-layer feed-forward ANN, the nodes in the input layer (input data introduced to the network) are linked with a predetermined activation function to those in the hidden layer, and then to the nodes in the output layer with a similar operation. An objective function with premise parameters (to be adjusted) is defined by comparing the difference between computed and target outputs. The fitness value of the objective function as well as associated parameters has to be adapted in the calibration process using optimization techniques. The optimal parameters are, therefore, determined corresponding to the most approximate computed outputs. Usually tan-sigmoid and linear functions are adopted as the activation functions to capture the relation of nodes between two layers. Nonetheless, they have no physical meanings and render the ANN a real 'black-box' model, and therefore, are unsuitable for the forecasting of river flows, due to the nonlinearity, non-stationary and seasonal behavior of river flows. It is essential to describe the hydrological processes through an adequate nonlinear and fuzzy model.

In the HNN model, the framework of the traditional ANN with three layers is maintained and, yet, activation functions with special significance are introduced. A conceptual function with fuzzy pattern-recognition idea from the input layer to the hidden layer is defined as follows

$$Q_i = \frac{1}{\sum_{l=1}^{C} \frac{\sum_{j=1}^{k}\left[w_{ji}\left(Q_j^{in} - M_i\right)\right]^2}{\sum_{j=1}^{k}\left[w_{ji}\left(Q_j^{in} - M_l\right)\right]^2}} \quad (1)$$

wherein Q_i ($i=1, 2, \ldots, s$) are nodes in the hidden layer and Q_j^{in} ($j=1, 2, \ldots, k$) denote nodes in the input layer. The parameter w_{ji} stands for weight parameter from the input layer to the hidden layer. A model vector is defined as $M=[M_i]=[M_l]$ in the hidden layer. The corresponding parameter C refers to the number of elements in the model vector. Note that C is the number of nodes in the hidden layer as well (i.e. $C=s$). Based on this activation function, the nodes in the hidden layers are elaborated by classifying the inputs into a number of categories. A higher value of C implies that there will be more categories in the hidden layer, which therefore represents a higher degree of nonlinearity of the model. In this paper, $C=11$ and $M=[1.0, 0.9, \ldots, 0.1, 0]$ are adopted