Journal of Hydrology 549 (2017) 603–615

Contents lists available at ScienceDirect

Journal of Hydrology

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

Research papers

A Pareto-optimal moving average multigene genetic programming model for daily streamflow prediction

Ali Danandeh Mehr [a,b,*], Ercan Kahya [a]

[a] Istanbul Technical University, Civil Engineering Department, Hydraulics Division, 34469 Maslak, Istanbul, Turkey
[b] Near East University, Department of Civil Engineering, P.O. Box: 99138, Nicosia, North Cyprus, Mersin 10, Turkey

ARTICLE INFO

Article history:
Received 12 January 2017
Accepted 5 April 2017
Available online 22 April 2017

Keywords:
Multigene genetic programming
Streamflow prediction
Pareto-optimal
Hydrological modelling

ABSTRACT

Genetic programming (GP) is able to systematically explore alternative model structures of different accuracy and complexity from observed input and output data. The effectiveness of GP in hydrological system identification has been recognized in recent studies. However, selecting a parsimonious (accurate and simple) model from such alternatives still remains a question. This paper proposes a Pareto-optimal moving average multigene genetic programming (MA-MGGP) approach to develop a parsimonious model for single-station streamflow prediction. The three main components of the approach that take us from observed data to a validated model are: (1) data pre-processing, (2) system identification and (3) system simplification. The data pre-processing ingredient uses a simple moving average filter to diminish the lagged prediction effect of stand-alone data-driven models. The multigene ingredient of the model tends to identify the underlying nonlinear system with expressions simpler than classical monolithic GP and, eventually simplification component exploits Pareto front plot to select a parsimonious model through an interactive complexity-efficiency trade-off. The approach was tested using the daily streamflow records from a station on Senoz Stream, Turkey. Comparing to the efficiency results of stand-alone GP, MGGP, and conventional multi linear regression prediction models as benchmarks, the proposed Pareto-optimal MA-MGGP model put forward a parsimonious solution, which has a noteworthy importance of being applied in practice. In addition, the approach allows the user to enter human insight into the problem to examine evolved models and pick the best performing programs out for further analysis.

1. Introduction

It is well documented that streamflow process is complex and not easily predictable (Yaseen et al., 2015). This is mainly due to the non-stationary feature of the phenomenon and highly nonlinear relationship between streamflow and the characteristics of its catchment (Nourani et al., 2011; Danandeh Mehr et al., 2013). One of the common ways to model streamflow process is to use of data-driven techniques, which have the ability to learn about and extract the nonlinear relationships between the streamflow and its driving variables. When using such techniques, a sound knowledge of the underlying physical processes is not prerequisite (Hundecha et al., 2001; Noori and Kalin, 2016).

Short-term streamflow prediction with a lead time less than (or equal to) one day is necessary for the real-time flood warning and reservoir operation systems (Danandeh Mehr et al., 2015). The application of various data-driven techniques, such as artificial neural networks (ANN), genetic programming (GP), and fuzzy logic in short-term streamflow prediction has been extensively evaluated and published in recent years (e.g., Hundecha et al., 2001; Moradkhani et al., 2004; Kücük and Agiralioglu, 2006; Wang et al., 2006; Makkeasorn et al., 2008; Shiri and Kisi, 2010; Kisi, 2010; Rezaeianzadeh et al., 2013; Krishna, 2013; Hosseinzadeh Talaee, 2014; Danandeh Mehr et al., 2015). Regardless of the type of the data-driven technique employed, prediction accuracy is highly dependent on the variables used to train and validate the technique. In most of the flow prediction studies on a short-term basis, daily rainfall and streamflow records have been used to create so-called rainfall-runoff models (e.g., Mutlu et al., 2008; Nourani et al., 2011, 2012; Shoaib et al., 2015). However, in poorly gauged basins, where no rainfall record is available, the rainfall-runoff commonly used models are not applicable. In such cases, single-station, cross-station, or successive-station runoff-runoff prediction models have been suggested (e.g., Ochoa-Rivera et al., 2002; Kisi and Cigizoglu, 2007; Demirel et al., 2009; Besaw et al.,

* Corresponding author at: Department of Civil Engineering, Near East University, P.O. Box: 99138, Nicosia, North Cyprus, Mersin 10, Turkey.

E-mail addresses: ali.danandeh@neu.edu.tr (A. Danandeh Mehr), kahyae@itu.edu.tr (E. Kahya).

http://dx.doi.org/10.1016/j.jhydrol.2017.04.045
0022-1694/© 2017 Elsevier B.V. All rights reserved.

2010; Can et al., 2012; Danandeh Mehr et al., 2015). For example, Kisi and Cigizoglu (2007) developed three ANN-based single-station prediction models to forecast daily streamflow at two rivers in Turkey. To this end, daily streamflow observations with one- to six-day antecedent records (lags) were considered as input vectors for the ANN. The authors demonstrated that three days lag (i.e., three input vectors) is sufficient to achieve the best one-day ahead streamflow forecasting model in respect of selected performance criteria. In another single-station streamflow prediction study, Özger (2009) developed two fuzzy inference systems using daily streamflow records at Demirkapi Station on Euphrates River, Turkey. Based upon strong serial dependence of observational flows, the author suggested that one- and two-day lags are enough to train the models. On the basis of auto- and partial autocorrelation analysis, Hosseinzadeh Talaee (2014) explained that one- to four-day lags of streamflow records at a station in Aspas Stream, Iran, are the most appropriate input vectors to train multilayer perceptron (MLP) neural networks for one-day ahead streamflow prediction. Most recently, Altunkaynak and Nigussie (2015) combined the SEASON algorithm with a MLP neural network and demonstrated that the new hybrid model can be used to extend lead time of single-station daily streamflow prediction models.

Despite providing acceptable prediction accuracy, none of the aforementioned studies provided explicit formulation in regard to single-station streamflow process. To bridge the gap, recent works have focused on applying GP to discover underlying process explicitly. Our review showed that only a few studies have investigated capability of GP in short-term streamflow prediction. For instance, Guven (2009) applied linear GP (LGP) and two versions of ANN to predict daily flow of Schuylkill River in the USA. The author demonstrated that the performance of LGP is higher than that of ANNs. Londhe and Charhate (2010) developed two GP-based one-day ahead streamflow prediction models at two stations in Narmada Catchment of India and demonstrated that GP performs superior than ANN and model trees. Although classical GP has been implemented in a few other streamflow modelling studies (e.g., Ni et al., 2010; Nourani et al., 2012, 2013b), at the best of our knowledge, no study has yet been conducted to assess the potential of multigene topology in GP, i.e., MGGP, for single-station daily streamflow prediction. Moreover, previous studies have investigated the effectiveness of GP mostly in terms of prediction accuracy and thus, additional studies are required to address problems associated with complexity of the proposed models. In this sense, this paper, for the first time, proposes a Pareto-optimal moving average multigene genetic programming (Pareto-optimal MA-MGGP) model to develop a parsimonious (accurate and simple) model for single-station streamflow prediction. The model is applied for daily streamflow prediction at Senoz Catchment in Turkey, and its performance is compared with those of stand-alone GP, MGGP, and conventional multivariable linear regression (MLR) prediction models as benchmarks. From practical point of view, the proposed model is explicit and parsimonious so that motivating to be used in practice.

2. Genetic programming (GP)

The state-of-the-art GP is of the most popular data-driven techniques that evolves computer programs to automatically solve problems using Darwinian natural selection (Koza, 1992). The task is done by randomly generating a population of computer programs and then breeding together the best performing programs to create a new population (offspring). Mimicking Darwinian evolution, this process is iterated until the population contains programs that solve the task well (Searson, 2015). In hydrological applications, GP is commonly used to infer the underlying structure of either natural (e.g., Ghorbani et al., 2010; Danandeh Mehr et al., 2013; Nourani et al., 2013b; Sattar and Gharabaghi, 2015; Meshgi et al., 2015; Ravansalar et al., 2016) or experimental (e.g., Selle and Muttil, 2011; Khan et al., 2012; Uyumaz et al., 2014) processes. In such applications, GP generates some possible programs (solutions) representing the underlying process mathematically. When the task is to build an empirical model of data acquired from a process or system, GP is often known as symbolic regression (Searson, 2015). GP is a self-structuring technique without requiring the user to know or specify the form of the solution in advance. It differs from either traditional regression analysis or other data-driven techniques, in which modeller must specify the structure of the given process. As shown in Fig. 1, potential programs are usually represented by tree structures with a root node, inner node(s), and leaves. Population of initial solutions is generated through a random processes such as *full*, *grow*, and *Ramped half-and-half* methods (Poli et al., 2008). Subsequent generations are commonly evolved through three genetic operators, namely reproduction, crossover, and mutation (Babovic and Keijzer, 2000). Reproduction is copying an existing population into the new population without alteration. Crossover is replacing the preferable parent's chromosome to produce an offspring and mutation is replacing a randomly selected functional or terminal node with same node from the preferable parents. An example of crossover and mutation operators to generate two new population (offspring) is presented in Fig. 2.

The major inputs for a standard GP modelling are: (i) patterns for training/validation; (ii) fitness function (e.g., mean square error) for tournament selection; (iii) functional (or inner nodes) and terminal (or leaves) sets for structural identification; and (iv) GP parameters for formation of a syntax tree (i.e., a program/potential solution). Depends on the degree of complexity of the process of interest, the functional set may contain the basic arithmetic operators (i.e., +, −, ×, ÷) or more complicated mathematical operators such as Sin, Exp, and others. In order to generate an initial program, an operator is chosen randomly from the predefined functional set to fill the root node. Then, inner nodes are filled randomly by a member of the either functional or terminal sets. Finally, leaves are filled by only predefined members of terminal set that may comprise independent variables, numerical/logical constants, or arguments for the applied functions. Owing to the probabilistic essence of GP approach, it is possible to derive a large number of potential solutions (functional expressions) for a problem. An experienced GP modeller can choose the best solution in different ways. For example, a Pareto-optimal solution (explained in Section 4) can be selected with respect to the both accuracy and complexity of potential solutions. Details on GP and its applications can be obtained from Gandomi et al. (2015).

2.1. Multigene genetic programming (MGGP)

In recent years, several advancements for classical GP (i.e., tree-based genotype also called monolithic GP) such as linear GP (LGP), multigene GP (MGGP), multi-expression programming (MEP), and gene expression programming (GEP) have been suggested. All these variants have a clear distinction in their genotype. A number of researchers have reported successful application of different genotype in GP (e.g. Brameier and Banzhaf, 2007; Guven, 2009; Danandeh Mehr et al., 2013, 2014a,b; Shoaib et al., 2015; Zorn and Shamseldin, 2015).

MGGP (Searson, 2009) is of the most recent advancements of GP that linearly combines low depth GP trees in order to improve fitness of classical GP. Owing to the use of smaller trees, the MGGP is expected to provide simpler models than those of classical GP (Searson, 2015). In MGGP, predictand variables are computed by