



Melody Search Algorithm Using Online Evolving Artificial Neural Network Coupled with Singular Spectrum Analysis for Multireservoir System Management

Farzane Karami¹ · Alireza B. Dariane¹

Received: 12 January 2020 / Accepted: 31 May 2021 / Published online: 12 June 2021
© Shiraz University 2021

Abstract

This paper introduces a novel application of online evolving neural networks (ENNs) trained by melody search algorithm (MSA) and singular spectrum analysis (SSA) preprocessing technique in finding optimum operating policies of Tehran water resources system with multiple reservoirs and multiple purposes. Harmony search (HS) algorithm is also used for comparison purposes. In addition, in order to better evaluate the proposed method, different combinations of artificial neural network (ANN) models are developed. It was noticed that the application of SSA preprocessing technique shows substantial improvement in the ANN model performance. In fact, application of singular spectrum analysis on the input data eliminates the irrelevant features and inverts the remaining components to yield a filtered time series. Moreover, all the results indicate the superiority of MSA over HS algorithm in all combinations of ANN models. It was also noticed that as the model becomes more complex, the MSA shows more improved and distinct results as compared to HS algorithm. Thus, it can be stated that the melody search is more efficient and a suitable surrogate for the harmony search algorithm in pursuing the objectives in complicated problems.

Keywords Melody search algorithm · Singular spectrum analysis · Artificial neural networks · Multi-reservoir system · Evolutionary optimization · Evolving ANN

1 Introduction

Reservoirs are probably the most important elements of complex water resources systems. The models expanded to provide operating rules for reservoirs are classified as simulation, optimization and combination of these two.

Over the past decades, engineers have paid much attention to the development of optimization models. As a consequent, many algorithms have been introduced to solve complicated engineering problems during past several decades. Most of these algorithms are based on numerical linear and nonlinear programming methods. In these methods, the results may depend on the selection of an initial point, if there is more than one local optimum in the problem. There was a turnover in early 1990s where evolutionary methods

were introduced. Then, in less than a decade, an explosion of research was directed toward development and application of these methods. This turnover was mainly due to simple structure, least knowledge of mathematics required, flexibility, and adaptability inherent in these heuristic methods.

Heuristic algorithms inspired mainly by natural systems are powerful tools developed in the last two decades to solve what was once considered as very hard to solve problem. There have also been methods that came to existence through inspirations by man-made systems and processes. Simulated annealing (SA), harmony search, and melody search algorithms are in this category. HS has gained an important momentum in the last few years.

Harmony search is an optimization algorithm that simulates the improvisation process of jazz music. It was first introduced by Geem et al. (2001) and later modified by other researchers. It has several impressive advantages, such as easy implementation, less adjustable parameters, and quick convergence, and has provided excellent results across different complex problems. The required memory of this algorithm is less than the other meta-heuristic methods.

✉ Farzane Karami
farzane_krm@yahoo.com

¹ Department of Civil Engineering, KN Toosi University of Technology, Tehran, Iran

Moreover, it does not need derivative information. On the other hand, the basic HS algorithm gets into trouble in performing local search for numerical applications. To resolve this issue, Mahdavi et al. (2007) introduced Improved harmony search (IHS) algorithm based on the basic one. The IHS algorithm has the power of the HS algorithm with the fine tuning feature of mathematical techniques and can outperform either one individually. Besides, Fesanghary et al. (2008) presented hybrid harmony search (HHS) algorithm to speed up local search. Other modifications on HS were carried out by Omran and Mahdavi (2008) and Pan et al. (2009, 2010). Ashrafi and Dariane (2013) introduced the melody search algorithm (MSA) in accordance with the concept of melody instead of harmony. They successfully applied this algorithm to various benchmark optimization problems and demonstrated the superiority of MSA over many other heuristic methods. This algorithm is based on musical performance processes, and interactive relations occurred between members of a group of musicians attempting to find better and better series of pitches within a melodic line. Moreover, the efficiency of the algorithm for solving shifted and rotated optimization problems is increased in this algorithm. The algorithm can better preserve the accuracy of the results comparing with other meta-heuristic methods in the case that the dimensionality of the problem or the entire feasible range of the search space is increased.

Parallel to heuristic algorithms, artificial neural network as another intelligent system was introduced in water resources in early 1990s. ANN is a modeling tool for recognizing arbitrary complex nonlinear relationships between input and output sets. It is increasingly being used to simulate and predict various water resources variables. Many studies focused on reservoir operation have shown that ANN is superior to the traditional techniques (Jain et al. 1999; Hasebe and Nagayama 2002; Chaves et al. 2004; Chaves and Chang 2008; Pianosi et al. 2011). Recently, ANNs have been also used in hybrid models for solving various water resources management problems (Wu et al. 2009; Ahmadi et al. 2013). ANNs are usually trained by backpropagation (BP) method. However, unlike its wide application, BP suffers from some drawbacks. Dariane and Karami (2014) introduced the single-step online evolving neural network to overcome some of these drawbacks. In most applications of ANN for deriving reservoir operation rule, the problem is handled following a two-step approach. In the first step, an optimization model is used to derive optimal target vector which is then set as the output target values in the ANN model in the second step. Backpropagation is based on the errors between ANN model output and the target output values. Therefore, in order to use BP, it is necessary to first identify the “optimal target vector” based on optimization methods. Meanwhile, it is inevitable to use long periods of data for optimization in order to develop adequate number of

target values for reliable training of the ANN model. Therefore, under these circumstances the optimization program would likely face dimensionality and computer run time problems. The single-step online approach overcomes these issues by eliminating the need for separate optimization and by directly using the ANN model to develop the operating policies through combining the two steps into a single one-step model.

Another drawback of BP is the restriction in using “the error indices minimization” as the objective function for the network training. In many problems, other types of objective functions such as maximizing system reliabilities or maximizing system benefits are preferred. Therefore, evolving ANN (i.e., ENN) was developed to overcome parts of BP drawbacks beside other merits.

The proper function of neural networks depends on many factors including the suitability of input variables and the amount of information they can add to the model in order to produce the required target output(s). Data preprocessing has been known as a successful approach for improving the value of input data in ANN models. Wavelet transform is a well-known and widely applied preprocessing method. Another and yet less applied approach is the singular spectrum analysis which is simpler but as powerful method for preprocessing the input data for ANN models.

The singular spectrum analysis was developed as the new time series method in 1970s and has been improved since then. SSA decomposes an original time series to trend, seasonal, semi-seasonal, and white. The new decomposed series help us to understand the trend of the original time series and to extract seasonal or monthly components and white noises (Myung 2009). SSA has been recognized as an appropriate preprocessing approach to couple with neural networks (or similar methods) for time series forecasting (Sivapragasam et al. 2001; Baratta et al. 2003; Wu et al. 2010; Wu and Chau 2011; Hassani and Mahmoudvand 2013). In recent years, SSA has not been only applied in the analysis of climatic and geophysical time series, but also in the analysis of social science and economic time series. This method has been used in various fields of research such as signal processing, nonlinear dynamics, medicine and mathematical statistics. Ghil and Taricco (1997), Danilov and Zhigljavsky (1997), Yiou et al. (2000) and Golyandina et al. (2001) published several papers dealing with methodologies and applications of SSA. The SSA approach can be also used as a forecasting algorithm for time series including hydrological series, or at least for some of its extracted components. Many models have been proposed for time series simulation and forecasting in hydrology such as auto-regressive moving average (ARMA) and auto-regressive integrated moving average (ARIMA) models. Unlike the traditional ARMA models, which require the data to be stationary and normally distributed, SSA has no restriction on the type of data. SSA does not depend on

any parametric model for the trend or oscillations and does not make any statistical assumptions concerning either signal or noise (Marques et al. 2006).

In this paper, in order to develop optimum operating policies for the large Tehran water resources system with multiple reservoirs and multiple purposes, several models were studied. In models one and two, we use harmony and melody search optimization algorithms to optimize releases and allocations from reservoirs to the demand sites in Tehran water resources system. These models try to explore the optimum path of releases throughout the whole period under the assumption that the time series of inflows are known in advance, meaning a perfect forecast. In models three and four, artificial neural networks which are trained by harmony and melody search algorithms are used to find optimum operation rule for each reservoir in the system. This is carried out in two stages. In the first stage, normalized data are used for training and testing and in the second stage, and the utilization of raw data in the neural network models was investigated. In models five and six, each inflow series are decomposed into components with the aid of SSA. The procedure is to decompose the original record first and then to build the forecasting model based on the decomposed series. A sensitivity analysis of the singular spectrum on lag parameter is carried out, and the appropriate number is chosen. Through these models, a complete and comprehensive comparison is made between different optimization methods and the power of MSA algorithm and the impact of using SSA are demonstrated. Since the different aspects of the algorithms have been studied simultaneously in various models, the obtained results are comprehensive and reliable.

2 Methodology

2.1 Evolutionary Algorithms

Meta-heuristic algorithms have been widely applied in recent decades to solve practical problems. Geem et al. (2001) developed a meta-heuristic algorithm that was conceptualized using the analogy of the music performance process named harmony search. The HS algorithm uses a stochastic random search that is based on the harmony memory considering rate and the pitch adjusting rate. Compared to earlier meta-heuristic optimization algorithms, the HS algorithm imposes fewer mathematical requirements. The parameters of HS include the harmony memory size (HMS), harmony memory consideration rate (HMCR), pitch adjusting rate (PAR), distance bandwidth (BW), and termination criterion. In harmony search algorithm, each solution is called a “harmony.” The “harmony memory” (HM) matrix is filled with randomly generated solution vectors at first and sorted in terms of the objective function value.

Then, a new harmony vector is produced based on three parameters including HMCR, PAR and BW. A good set of parameters can enhance the algorithm’s ability to search for the global optimum. The following general steps are taken in applying the HS (Dariane and Karami 2014). First of all, if a uniform random number returned by $rand()$ in $[0,1]$ is less than $HMCR$, the decision variable is generated by the memory consideration; otherwise, it is obtained by a random selection. Secondly, each decision variable updated by the memory consideration undergoes a pitch adjustment with a probability of PAR . Thus, every component obtained by the memory consideration is examined to determine whether it should be pitch-adjusted. This operation uses the PAR parameter. In the memory consideration, new harmony is chosen from harmony memory. And finally, new harmony is produced by random selection. In a nutshell, the scheme to improvise a new harmony, X_{new} , can be summarized as follows (Dariane and Karami 2012).

```

For (j=1 to n) do
    If( $r1 < HMCR$ ) then
         $x_{new}(j) = x_a(j)$ 
        if ( $r2 < PAR$ ) then
             $X_{new}(j) = X_{new}(j) \pm r * bw, r \in (0,1)$ 
        End if
    Else
         $X_{new}(j) = LB(j) + r * (UB(j) - LB(j)), r \in (0,1)$ 
    End for

```

If the objective function of the new harmony vector is better than the worst harmony in the HM, the new harmony is included in the HM and the existing worst harmony is excluded from the HM. Then, the harmony memory is sorted again. This process is continued until stopping criterion is obtained (Mahdavi et al. 2007; Lee and Geem 2005).

HS has been able to attract many researches for developing HS-based solutions of different engineering and optimization problems. Although the algorithm is computationally effective and easy to implement for solving various kinds of engineering optimization problems, it is not quite successful in performing local search in continuous numerical optimization applications (Mahdavi et al. 2007), especially for problems with high dimensions. Consequently, advantages of this algorithm led the researchers to improve the performance and develop further applications with different ideas. Several attempts have been made to improve the performance of basic HS algorithm. Improved harmony

search algorithm (IHS) which was developed by Mahdavi et al. (2007), dynamically updates the values of PAR and BW as follows:

$$\text{PAR}(\text{gn}) = \text{PAR}(\text{min}) + \frac{\text{PAR}(\text{max}) - \text{PAR}(\text{min})}{\text{NI}} * \text{gn} \quad (1)$$

$$b_w(\text{gn}) = b_{w,\text{max}} * \exp(C.\text{gn}) \quad (2)$$

$$C = \frac{\ln\left(\frac{b_{w,\text{min}}}{b_{w,\text{max}}}\right)}{\text{NI}} \quad (3)$$

Omran and Mahdavi (2008) proposed global-best harmony search (GHS) algorithm by modifying the pitch adjustment rule. Fesanghary et al. (2008) combined the two powerful search algorithms, namely the sequential quadratic programming and the harmony search algorithm, and proposed a new method named hybrid harmony search (HHS). Then, the EHS algorithm was proposed by Pan et al. (2010) to self-adaptively determine the best parameter set for different search phases during the evolution process. Inspired by the GHS algorithm, a self-adaptive GHS (SGHS) algorithm was presented by Pan et al. (2009). In short, the new scheme to improvise a new harmony, X_{new} , can be summarized as follows (Pan et al. 2009):

```

For (j=1 to n) do
    If (r1 < HMCR) then
        xnew(j) = xa(j) ± r * bw, r ∈ (0,1)
        if (r2 < PAR) then
            Xnew(j) = XB(j)
        End if
    Else
        Xnew(j) = LB(j) + r * (UB(j) - LB(j)), r ∈ (0,1)
    End for

```

In this algorithm, it is assumed that the HMCR and PAR values are normally distributed in the range of [0.9, 1.0] and [0.0, 1.0] with mean 0.98 and 0.9 and standard deviation 0.01 and 0.05, respectively. During the evolution, the values of HMCR and PAR associated with the generated harmony successfully replacing the worst member in the HM are recorded. After a specified number of generations LP, the mean values are recalculated by averaging all the recorded values during this period. With the new mean and the given standard deviation, new HMCR and PAR values

are produced and used in the subsequent iterations. The above procedure is repeated to find the solution. Moreover, it should be noted that the value of b_w decreases dynamically with increasing generations (NI) as follows (Pan et al. 2009):

$$b(t) = \begin{cases} b_{w,\text{max}} - \frac{b_{w,\text{max}} - b_{w,\text{min}}}{\text{NI}} * 2t, & \text{if } t < \text{NI}/2 \\ b_{w,\text{min}}, & \text{if } t > \text{NI}/2 \end{cases} \quad (4)$$

Ashrafi and Dariane (2013) introduced melody search algorithm. Although melody search adopts the basic concepts of HS, the structure is quite different. This algorithm simulates the musical performance processes of a musician group. As it was previously mentioned, there is one memory in HS named harmony memory (HM) and all players sound a harmony together in each step. In engineering optimization, each musician represents a decision variable, and its preferred sound pitches are the preferred values of variable. Unlike harmony search algorithm, in melody search algorithm, melody memory (MM) consists of several player memories (PM). In a group of musicians, each music player has a specified memory and sounds a series of pitches within the possible ranges. Figure 1 demonstrates the structure of melody memory. This algorithm consists of two different phases. In the initial phase, each player improvises his/her melody individually. In this phase, players do not influence each other. In the next phase, the new possible range for each variable is calculated from the best melody of each player memory PM. For this purpose, variables of the best melody of each PM are saved after updating player memories. The minimum and maximum ones for each variable specified the new possible. Hence, best bounded ranges of each parameter specified in the group that consists of several different melodies with different musical qualities, while these ranges are changed through different iterations (Ashrafi and Dariane 2013). This process is shown in Figs. 2 and 3.

Melody search parameters that must be defined at first are the number of player memories (PMN), player memory size (PMS), maximum number of iterations (NI), maximum number of iterations for the initial phase (NII), distance bandwidth (b_w), minimum and maximum values of pitch adjusting rate (i.e., PAR_{min} and PAR_{max}), and player memory considering rate (PMCR).

2.2 Online Evolving Artificial Neural Network

Artificial neural networks refer to computing systems whose central theme is borrowed from the analogy of biological neural networks. An ANN contains a number of neurons that are arranged in an input layer, an output layer, and one or more hidden layers. The number of input and output nodes is dependent on the problem to which the network is being applied. Two hidden layers with sufficient numbers of neurons would be adequate in any problem (Dariane and

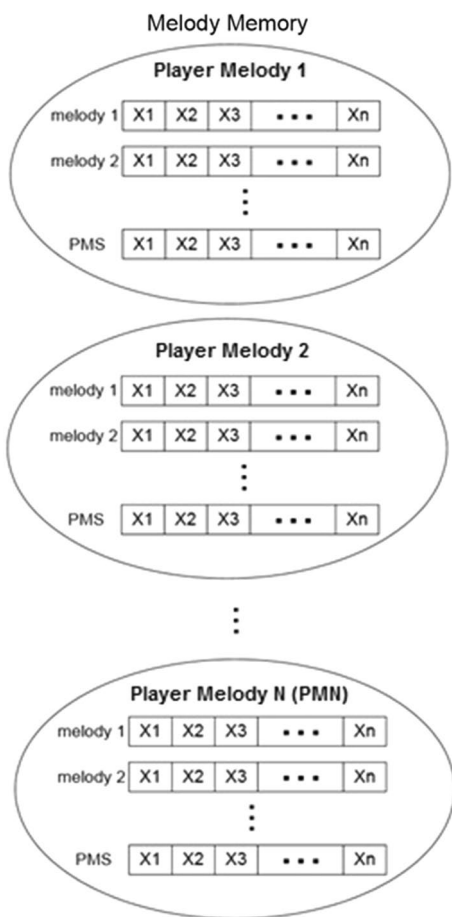


Fig. 1 Structure of melody memory

Moradi 2014). However, one hidden layer provided there are enough neurons in the hidden layer could handle most cases. A network learns by adjusting the biases and weights that link its neurons. Backpropagation algorithm is currently the most common approach to train feed forward networks. In this algorithm, a training pair is selected from training set and applied to the network. The output is calculated and compared with the expected outputs (targets) identified by the training pair. The weights and biases of each neuron are adjusted afterward by the differences between the expected network and actual outputs. As mentioned earlier, algorithms like BP are limited to certain conditions and constraints. Hence, an optimization-simulation procedure is needed where different objective functions and system constraints could be easily handled. Dariane and Karami [15] introduced an approach called the single-step online evolving neural network that employs a heuristic algorithm to train the neural network. Online ENN can handle any forms of objective function and has overcome the restrictions of BP method. In addition, according to Dariane and Moradi [29] this approach also avoids over-fitting which is a major issue with ANNs using BP method and therefore does not require validation period to keep the model from over-fitting. Thus, the validation period data can be used for better training the network. Moreover, unlike commonly practiced two-step methods inherent in ANNs using BP applied for large reservoir operation problems with known issues such as extensive computer time consumption and dimensionality, the single-step online ENN can easily handle large multiple reservoir systems.

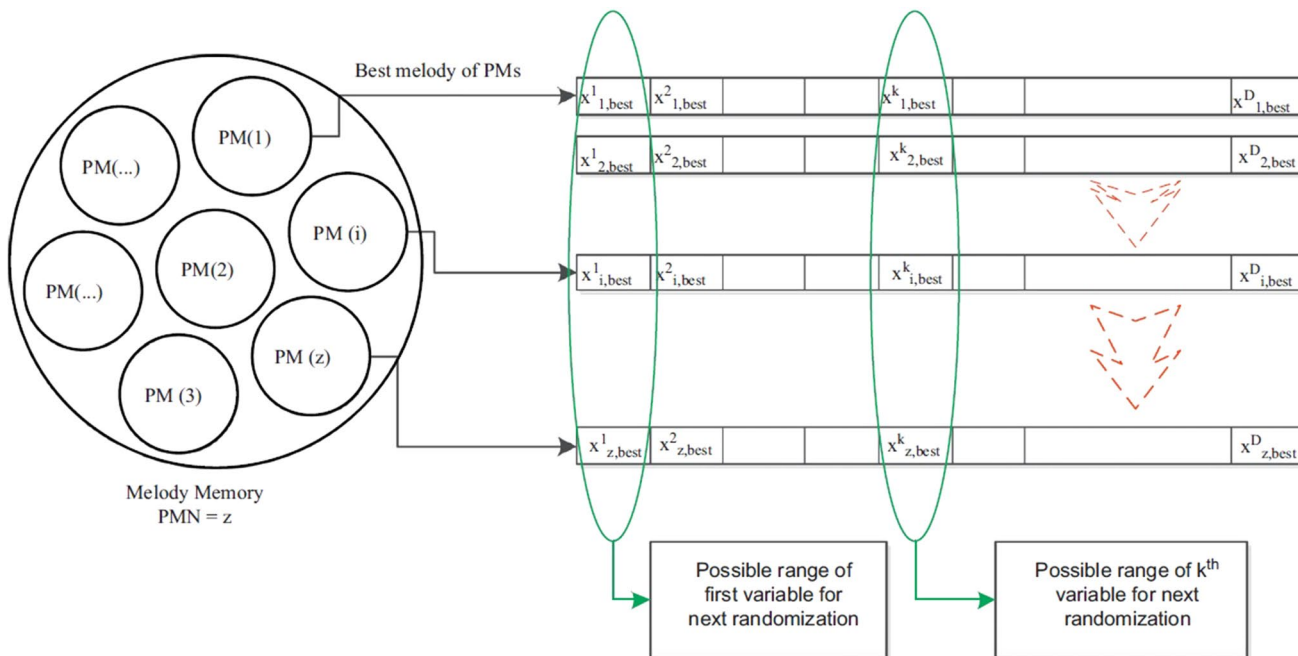


Fig. 2 Melody search algorithm (Ashrafi and Dariane 2013)

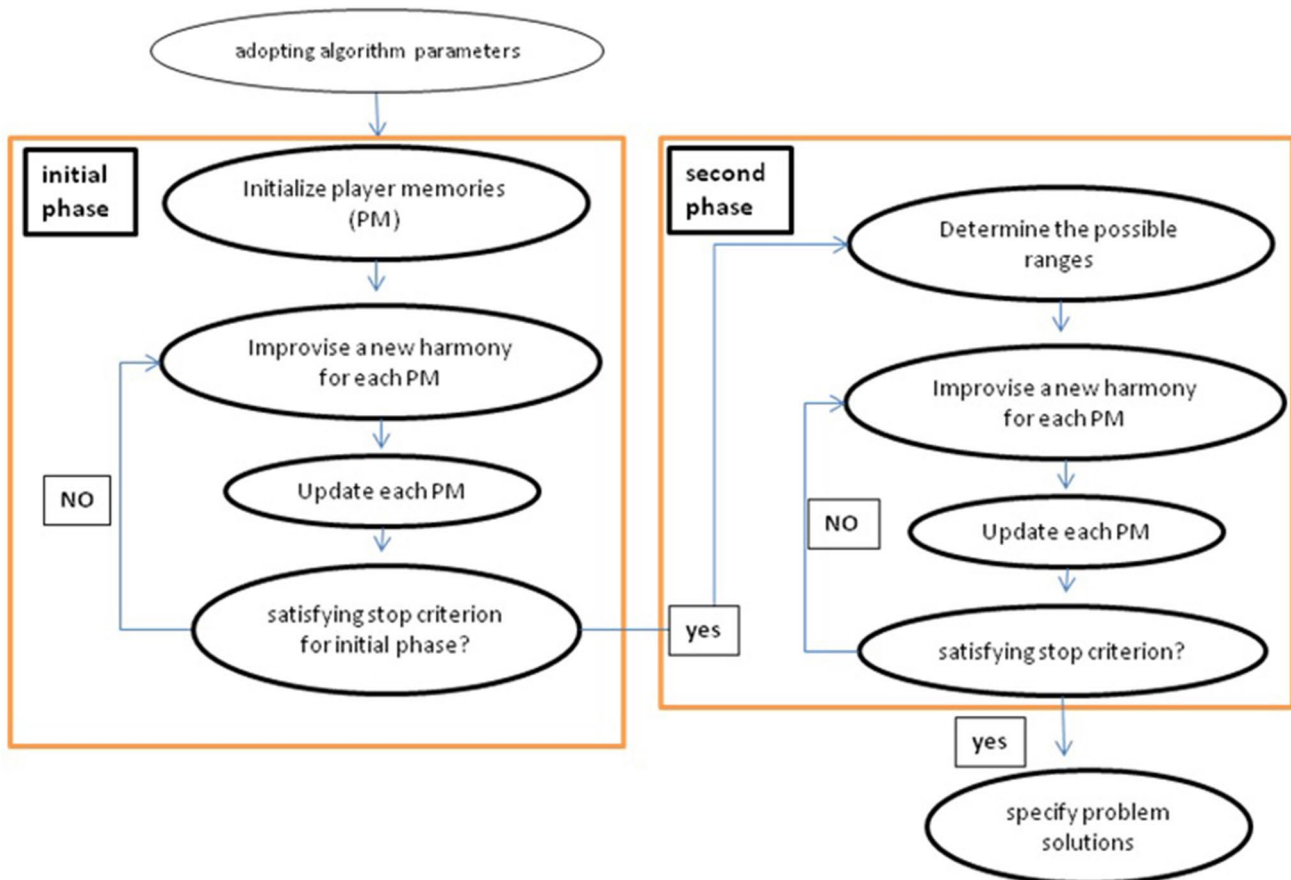


Fig. 3 Flowchart of melody search algorithm

2.3 Singular Spectrum Analysis

Singular spectrum analysis is generally seen as an adaptive noise-reduction algorithm which is widely used in digital signal processing. SSA method, unlike the wavelet transform, is very simple and could be easily programmed. SSA is also used as an efficient preprocessing algorithm. Preprocessing methods extract the hidden useful information within the time series and by introducing them as inputs to the neural networks, improve the training and hence the model performance. SSA is also a very useful tool for finding trends of different resolution, smoothing, extraction of seasonality components, simultaneous extraction of cycles with small and large periods, extraction of periodicities with varying amplitudes, simultaneous extraction of complex trends and periodicities, finding structure in short time series (Hassani and Mahmoudvand 2013). The basic SSA algorithm has two stages: decomposition and reconstruction. The decomposition stage requires embedding and singular value decomposition (SVD). Embedding decomposes the original time series into the trajectory matrix; SVD turns the trajectory matrix into the decomposed trajectory matrices which will turn into the trend, seasonal, monthly components, and white noises according to their singular

values. The principal components (PCs) are a projection in a different coordinate system, and hence, their interpretation is different from the original time series. Hence, the original time series $X(t)$ and the principal components cannot be compared. For comparison the reconstruction stage is required. However, by projecting the PCs back onto the eigenvectors, we obtain time series in the original units. This step can be also skipped if one does not want to precisely extract hidden information by regrouping and filter of components (Wu and Chau 2011). The decomposition stage requires the following steps:

2.3.1 Embedding

In the first step initial time series change into a matrix (Y) that contains the original time series in the first column, a lag-1 shifted version of that time series in the second column, etc. For example, as illustrated in the following, if the “window size” is $M=4$, i.e., only lags of $k=0, 1, 2$ and 3 are considered.

$$Y = [x(t), x(t+1), \dots, x(t+M)]$$

$$Y = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ a_2 & a_3 & a_4 & a_5 \\ a_3 & a_4 & a_5 & 0 \\ a_4 & a_5 & 0 & 0 \\ a_5 & 0 & 0 & 0 \end{bmatrix}$$

2.3.2 Computing Covariance Matrix

The covariance matrix C is computed as $C = Y^*Y/N$, which Y^* is transpose of Y and N is number of data points. This equation follows from the definition of covariance for the case where the time series X has a mean and variance equal to 0 and 1, respectively.

2.3.3 Computing Eigenvectors of the Matrix C

The eigenvectors of the matrix C are computed, and matrix RHO is constructed from these eigenvectors. The columns of the matrix RHO are the eigenvectors. The eigenvectors of the matrix C tell us something about the temporal covariance of the time series, measured at different lags,

2.3.4 Principal Components

The principal components are computed by following simple matrix equation. The principal components are again time series, of the same length as the “embedded” time series. Each column of the matrix PC is the principal component $PC1, PC2, PC3$, etc.

$$PC = Y^*RHO \tag{5}$$

This step can be also skipped if one does not want to precisely extract hidden information by regrouping and filter of components.

3 Application

3.1 Case Study

Tehran is the capital of Iran and with nearly thirteen million populations is considered a megacity. It is located in northern Iran at 35°45' northern latitude and 51°30' eastern longitude. Water resources in this region include Taleghan, Karaj, Lar, Latian and Mamloo reservoirs and a system of wells scattered in the region. Municipal and agricultural demands of the city are estimated from the previous studies and are shown in Table 1 along with long-term monthly average inflows at different sites. Figure 4 schematically shows the location map of the system under study. The region has three agricultural sites, namely Taleghan, Karaj and Varamin. Lar reservoir suffers from excessive water escape and is unable to store water. Therefore, in this site available water in each time period is transferred to Tehran using the maximum channel capacity and the remaining is transferred to Latian reservoir, again up to its maximum channel capacity. Moreover, Taleghan and Karaj reservoirs act in parallel in meeting the municipal demand of Tehran. Similarly, Latian and Mamloo act as parallel in meeting Tehran water demand, although they are in cascade on Jajrud river. Groundwater withdrawal in this study is restricted to 120 million cubic meters (mcm) annually as the first priority; however, in drought situations withdrawals up to 250 mcm are allowed as the second source priority for meeting the municipal demands of the capital. Orders of the months are based on water year with October as the first and September as the last month in each water year.

The models are prepared and tested using 47 years of monthly measured data (1958–59 to 2004–05 water years). The first 35 years of monthly data are used in optimization mode for deriving the operation rules and the remaining 12 years are used in simulation mode for testing the models.

Table 1 Average inflows and target demands (million cubic meters, mcm)

	Reservoir	Month												Annual
		1	2	3	4	5	6	7	8	9	10	11	12	
Inflows	Taleghan	10.4	14	12.7	11.5	12.9	24	78.7	138.6	99.2	39.1	16.7	10.4	468.2
	Karaj	13	14.6	13.5	12.3	13	22.8	57.5	92.5	78.4	45.4	24.4	16.3	403.8
	Latian	9.4	13.3	13	12.7	15	30	72.8	84.3	48.8	21	11.6	9.1	341.2
	Mamloo	3.8	5.5	5.7	5.2	6.4	12.8	28.6	33.8	18.7	7.2	4	3.1	134.8
	Lar	14.6	13	11.1	9.6	8.8	10.4	39.9	110.3	105.5	50.6	28.9	20.5	423.3
Demands	Tehran Municipal	85.3	79.9	77.8	77.5	78	81.3	76.8	84.2	92.6	100	101	96.2	1030
	Taleghan ag	10	3.5	0	2.8	2.9	7.1	27.7	41.4	54.4	65.8	61.6	32.8	310
	Karaj ag	3	1	0	0.8	0.9	2.1	8.3	12.3	16.2	19.6	18.4	9.8	92.4
	Varamin ag	5.3	6.3	3.4	3.7	7.9	15.8	40.4	56.6	51.9	50.6	34.9	17.8	294.6

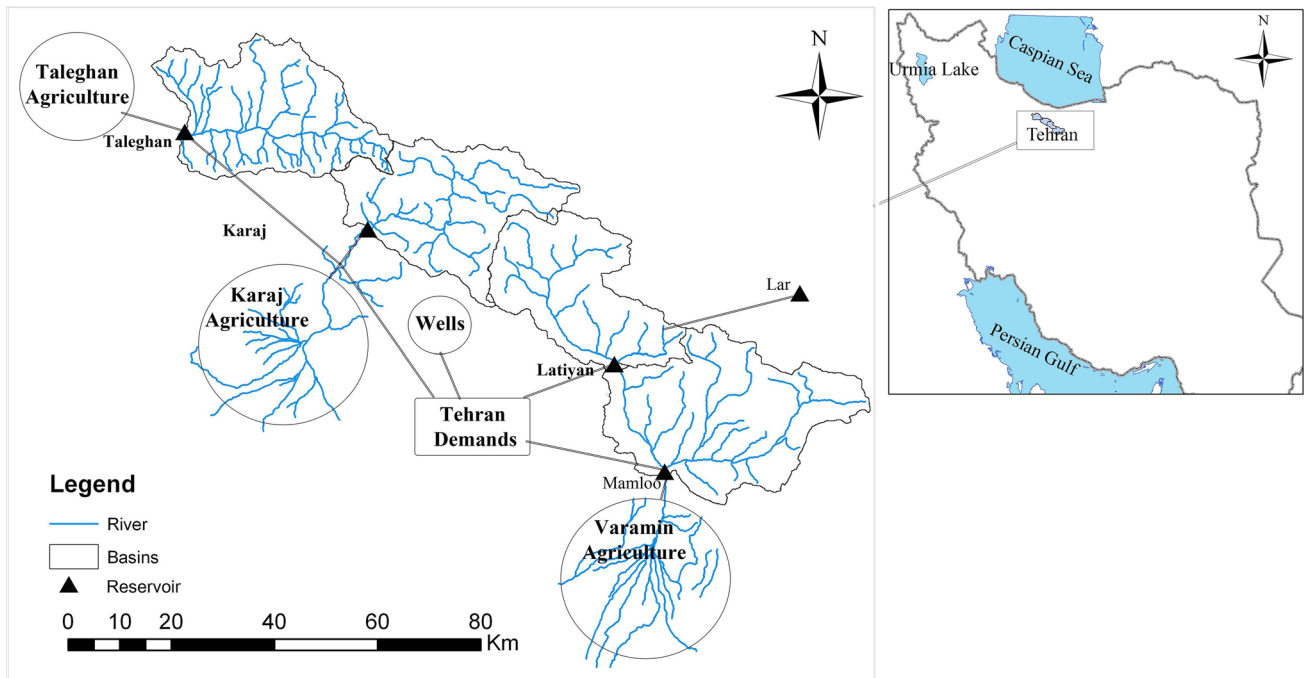


Fig. 4 Tehran location map

Equation (6) is used as the objective function where y = year index; t = month index; c = index of agricultural demand sites; $TD_{teh(y,t)}$ = Tehran municipal demand; $R_{teh(y,t)}$ = sum of releases to Tehran from all reservoirs and groundwater; $TD_{agr(y,t,c)}$ = agricultural demand and $R_{agr(y,t,c)}$ = releases to agricultural sites.

$$\text{Minimize } Z = \sum_{y=1}^n \sum_{t=1}^T (TD_{teh(y,t)} - R_{teh(y,t)}) + \sum_{y=1}^n \sum_{t=1}^T \sum_{c=1}^3 (TD_{agr(y,t,c)} - R_{agr(y,t,c)}) \quad (6)$$

3.2 Models

In the following sections, models developed in this study are briefly explained.

3.2.1 Modeling Release Optimization with Perfect Forecast

In models one and two, we use harmony and melody search optimization algorithms to optimize releases and allocations from reservoirs to the demand sites in Tehran water resources system. These models try to explore the optimum path of releases throughout the whole period under the assumption that the time series of inflows are known in advance, meaning a perfect forecast. This is accomplished without considering any types of system operation rule. Therefore, the whole data are utilized by the models for finding the optimum path. Obviously, here there is no need to set two periods of calibration and test for the models.

3.2.2 Modeling Online Evolutionary ANN

In models three and four, artificial neural network is used to find optimum operation rule for each reservoir in the system. The neural network considered for these models has a single hidden layer, and the number of hidden neurons was found to

be 2 using a trial and error method. Input layer contains three neurons, and the output layer contains one neuron. Inflows and demands are seasonal with a pronounced annual cycle. Hence, additional neurons to consider seasonal information are not needed. The networks for model three and four are trained by harmony and melody search algorithms, respectively. In these models, reservoir operation rule curves are established by the neural networks with weights and biases as the decision parameters. Thus, a portion of data is used to derive the ANN-based rule curves and a smaller portion is set aside to test the model performance using the calibrated rule curve parameters (i.e., weights and biases).

The study is carried out in two stages. In the first stage, normalized data are used for training and testing. The sigmoid and linear transfer function is used in hidden and output layers, respectively. In the second stage, the utilization of raw data in the neural network models was investigated. The results were far from satisfactory in this stage indicating

that normalization helps for better performance of the ANN models. It is worthwhile to mention that when raw data are used the transfer function in hidden and output layers must be linear. Otherwise, there would be a scale discrepancy in the outputs. For example, if a sigmoid function was used, it would transform the data to a range between 0 and 1, which is not suitable for comparison with the raw output targets. Nevertheless, although the network tries to overcome the problem by stretching further the weights and biases, the overall performance falls beyond satisfactory levels. However, there are cases, where raw data produce more satisfactory results than normalized data (see Dariane and Moradi 2014).

Based on our assumption, the output neuron indicates the end of period reservoir storage ($S_{(y,t+1)}$). Then, the amount of release is calculated from the mass balance equation as follows.

$$R_{total(y,t)} = S_{(y,t)} + Q_{(y,t)} - S_{(y,t+1)} \tag{7}$$

where $S_{(y,t)}$ is the storage level at the beginning of period t in year Y . $R_{total(y,t)}$ and Q_t are the total release and inflow during period t , respectively. Although we neglected losses such as evaporation, these losses can be easily incorporated into the mass balance equation without any change in the methodology. The reader is referred to the work by Dariane and Karami (2014) for more detailed discussion on application of ENN models in water resources systems.

3.2.3 Modeling ENN Coupled with SSA

In models five and six, each inflow series are decomposed into components with the aid of SSA. The procedure is to

decompose the original record first and then to build the forecasting model based on the decomposed series. The decomposition by SSA requires identifying lagged number. A sensitivity analysis of the singular spectrum on lag parameter is carried out and the appropriate number is chosen. The decomposed inflows along with some other time series are set as inputs to the network (Fig. 5). The number of hidden neurons was found to be 4 using a trial and error method. In models five and six, the networks are trained by harmony and melody search algorithms, respectively.

4 Discussion of Results

In order to develop optimum operating policies for Tehran water resources system with multiple reservoirs and multiple purposes, several models were studied. Results are orderly presented in the following section. Note that at the beginning in all the models, initial storage in the first training period of all reservoirs are assumed to be at 90 percent of the maximum storage levels. The objective function in all the models is to minimize the total deficit as defined in the previous section. As it was mentioned earlier, the models are prepared and tested using 47 years of monthly measured data (1958–59 to 2004–05 water years). The first 35 years of monthly data are used in optimization mode for deriving the operation rules, and the remaining 12 years are used in simulation mode for testing the models. It should be noted that all models convergence to steady condition after about 1,500,000 iterations. Hence, the search process could have been stopped in that iteration. However, the search was allowed to continue up to 3,000,000 iterations in all models, to see if any considerable changes would take place after

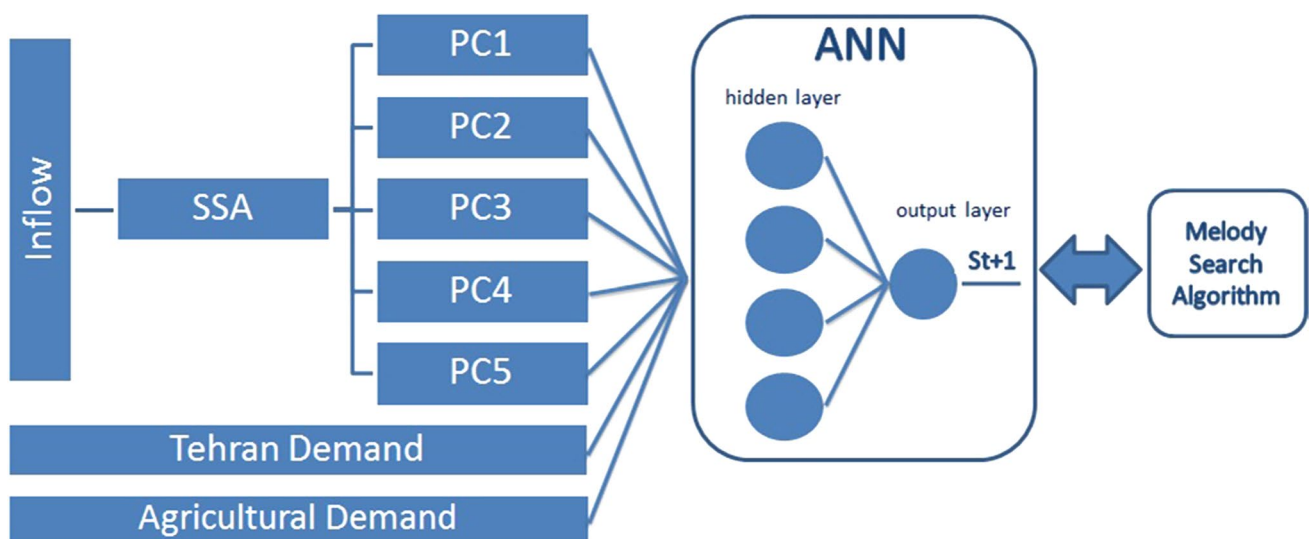


Fig. 5 Modeling single-step online ENN with SSA

the initial convergence. Other parameters were obtained by sensitivity analysis.

4.1 ENN with Raw or Normalized Data

In most applications in the past, neural network modeling has been used with normalized data. Unfortunately, they do not discuss the reason for following such an approach, neither they mention why the raw data were not used instead. Here, we first discuss the normalization versus raw data approach and whether or not normalization must be always assumed in ANN applications. Then, the utilization of SSA is discussed.

Table 2 shows the results of applying ENN in Tehran water resources system when raw and normalized inputs are used. Reliability is defined as the probability that the reservoir will perform the required demand and vulnerability is defined by Eq. 8.

$$\text{Vulnerability max} = [\text{Max (Demand – Release)}] / \text{Demand} \tag{8}$$

The results show that ENN based on normalized data outperforms the model based on raw data. Therefore, normalization is applied in all the following models in this study. As it was mentioned earlier, poor performance of models based on raw data could be due to scale discrepancies. For example, large input values into an ANN would require extremely small weighting factors to be applied and this can cause a number of problems including inaccuracies introduced by floating point calculations (Bhawan 2000).

However, in basins that maximum and minimum of events are not known, using ENN with raw data instead of normalization may yield better results. The reason is that for normalizing the data, maximum and minimum of events must be determined. To cope with this issue, they can be assumed based on observations at hydrometric stations or previous experiences in the basin. These assumptions can cause approximation and thereby may decrease the whole system performance. Yet, since they are known in this application, utilization of ENN with normalized data is preferred over raw data approach.

Table 2 Comparison of ENN performance with raw or normalized data over the testing horizon

	Deficit			Max. vulnerability		Reliability
	Drinking	Agri	All	Drinking	Agri	
Raw data	494	5491	5985	0.49	1	0.52
Normalized data	303	5313	5616	0.44	1	0.64

4.2 Sensitivity Analysis of the Singular Spectrum on Lag Parameter

As it was mentioned earlier, the decomposition by SSA requires identifying lagged number. A sensitivity analysis of the singular spectrum on lag parameter is carried out, and the appropriate number is chosen. Table 3 shows the results of sensitivity analysis of the singular spectrum on the lag time. As it can be seen from this table, the results can be improved by increasing the number of lags up to four. After that no improvement is observed, however, more computation load would be added. Therefore, the final lag parameter in SSA was set as four (i.e., five PCs).

SSA approach is based on the idea that the predictability of a system can be improved by revealing the important oscillations in time series taken from the system. For instance, Fig. 6 shows the five PCs obtained by SSA decomposition for Karaj inflow. It can be seen that the SSA method decomposes the time series into the number of components with simpler structures, such as a slowly varying trend, oscillations and noise. As explained above, the principal components of the time series are constructed by using the eigenvectors. The eigenvectors tell us something about the temporal covariance of the time series, measured at different lags. The principal components are again time series, of the same length as the “embedded” time series (i.e., the matrix Y). The difference between PC and Y is that the columns of PC do not correspond to different time lags. However, the original values of Y have been transformed (i.e., projected in a new coordinate system) in order to contain most of the variance in the first PC, most of the remaining variance in the second PC, and so on. It should be noted that the PCs are “orthogonal” at lag zero, i.e., there is no covariance between the PCs (although there is covariance between the PCs at nonzero lags). This shows that the variance of each PC is

Table 3 Sensitivity analysis of SSA on the lag time

Lag time	Reliability		Max. vulnerability	
	Drinking	Agri	Drinking	Agri
1	0.64	0.41	0.44	1
2	0.66	0.43	0.44	1
3	0.68	0.43	0.43	1
4	0.71	0.43	0.4	1
5	0.71	0.43	0.4	1

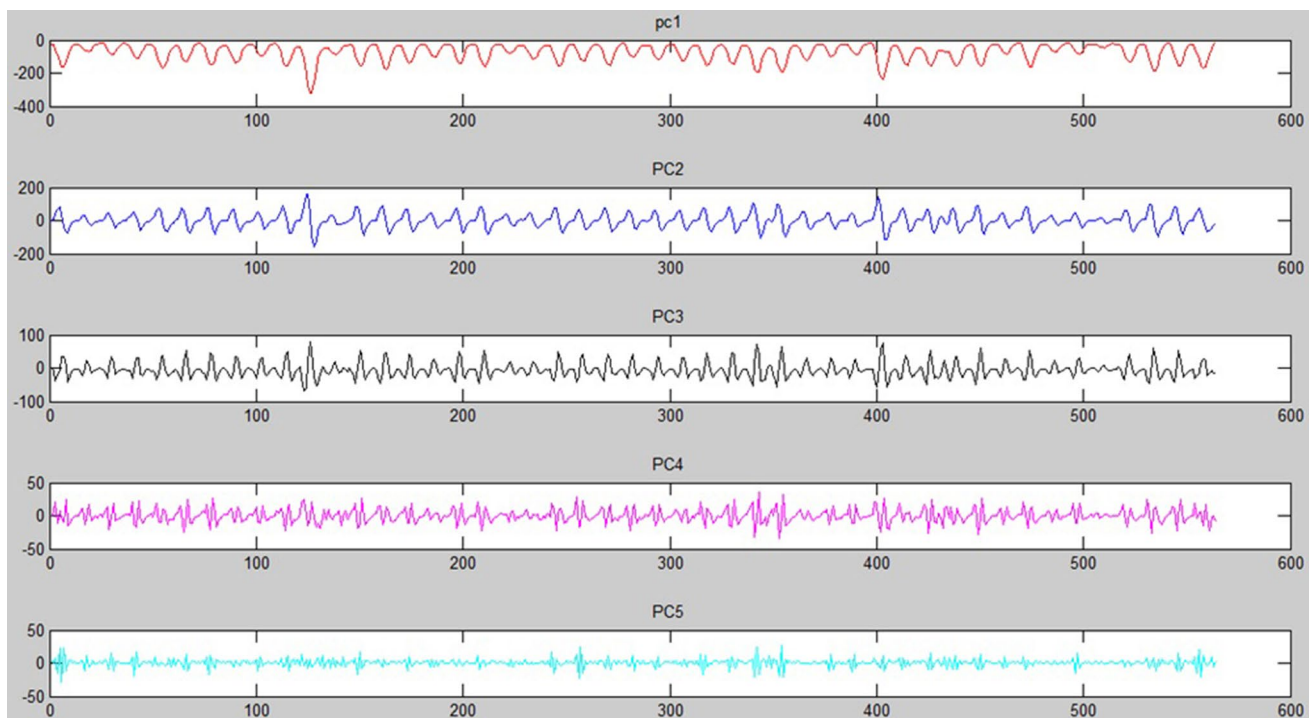


Fig. 6 PCs of Karaj inflow produced by SSA

equal to the eigenvalue of the corresponding eigenvector. Moreover, there is no covariance between the PCs (at lag zero). Theoretically, the diagonal should contain the eigenvalues, and the off diagonal elements should all be zero. As it was mentioned above and it can be seen from Fig. 6, the last PC of Karaj inflow contains very little variance, a fact that we already knew from the eigenvalues, and the first PC accounts for the maximum variance. The rest of the series are likewise.

4.3 Comparison of Models

Table 4 shows the results of applying different models in Tehran system. As it was mentioned earlier, models

one and two are regular optimization models with perfect inflow forecast assumption and do not use any type of operation rule, including ANN, in their structures. Therefore, the results obtained by these models could be assumed as the best possible performance of the system in case if rule curves were applied. However, interestingly as it can be seen from the results of objective function, vulnerability and reliability of models three and four, ANN-based operation rules developed in these models even performs better than unrestricted and rule-free models (i.e., models one and two), which is an indication of the power of artificial neural networks on mapping nonlinear and complicated system problems. As it was mentioned earlier, models three and four use operation rules based on

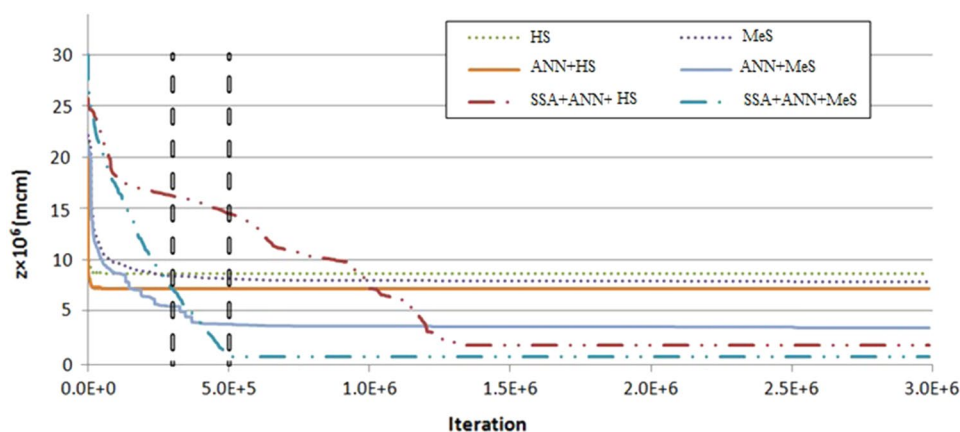
Table 4 Comparison of optimization models during test period

Category	#	Method	$z \times 10^6$	Reliability		Max. vulnerability		Deficit (mcm)	
				Drinking	Agri	Drinking	Agri	Drinking	Agri
Optimization with perfect forecast	1	HS*	0.07	0.64	0.4	0.45	1	311	5278
	2	MSA*	0.06	0.64	0.4	0.43	1	308	5108
Online evolutionary ANN	3	ANN, HS	0.05	0.64	0.4	0.44	1	303	5313
	4	ANN, MSA	0.03	0.69	0.42	0.4	1	250	5531
ENN coupled with SSA	5	SSA, ANN, HS	0.02	0.71	0.43	0.4	1	278	4828
	6	SSA, ANN, MSA	0.01	0.75	0.43	0.39	1	243	4366

Bold sections are types of methods

*includes the whole period

Fig. 7 Convergence function of different models



artificial neural network trained by harmony and melody search algorithms.

And finally, as it can be seen from Table 4 the application of SSA preprocessing on input data shows a distinct improvement in the objective function as well as other criteria including reliabilities and deficits. Model six decreases the objective function from 0.06 and 0.03 to 0.01 which is a significant reduction. Moreover, the sixth model (ANN coupled with SSA and MSA) resulted in lowest water deficit, 243 mcm for drinking water and 4366 mcm for agriculture when compared with other methods.

In the following, we show that melody search is more efficient and a suitable surrogate for the harmony search algorithm in pursuing the objectives in a complicated multiple reservoir system such as the one used in this research. Figure 7 displays convergence function. In order to compare convergence rate, two sections in iteration 300,000 and 500,000 are used for further analysis.

As it can be seen from Fig. 7, the difference between harmony and melody search performance is increased as the models become more complicated. The largest difference is observed in models five and six where ANN and SSA are included in the model. In other words, the rate of convergence of the models using MSA increases in comparison with HS-based models as more complication is introduced into the models. On the other hand, as it can be seen from figure, if we had a cost obligation to cut the iterations in earlier stages (i.e., in 300,000 or 500,000) rather than continuing until 3,000,000 iterations, the MSA-based models would have performed much better than those using HS algorithm. Therefore, although HS itself has shown to be a promising algorithm among well-known heuristic methods, the results here show that the novel melody search algorithm is more

powerful and reliable than HS in complicated and large-scale water resources problems.

5 Conclusions

This paper examined the development and application of melody search algorithm in a multiple reservoir, multiple purpose large-scale water resources system in Tehran, Iran. The evaluation was carried out by a comparative study between MSA and HS algorithms. Comparison of results of the six models developed in this research indicated the distinguished power of melody search algorithm over harmony search. Results displayed that this algorithm is very successful for complicated and large-scale problems where the dimensionality and computer run time are the main issues. Moreover, it was shown that results could be further improved by decomposition of inflow series by singular spectrum analysis.

References

- Ahmadi MA, Zendejboudi S, Lohi A, Elkamel A, Chatzis I (2013) Reservoir permeability prediction by neural networks combined with hybrid genetic algorithm and particle swarm optimization. *Geophys Prospect*. <https://doi.org/10.1111/j.1365-2478.2012.01080.x>
- Ashrafi SM, Dariane AB (2013) Performance evaluation of an improved search algorithm for numerical optimization: Melody Search (MS). *Eng Appl Artif Intell* 26(4):1301–1321. <https://doi.org/10.1016/j.engappai.2012.08.005>

- Baratta D, Cicioni G, Masulli F, Studer L (2003) Application of an ensemble technique based on singular spectrum analysis to daily rainfall forecasting. *Neural Netw* 16:375–387
- Bhawan J., 2000, “ Application of artificial neural networks (ANN) in reservoir operation”, National Institute of Hydrology, Uttaranchal, India, TR/BR-6/1999–2000
- Chaves P, Chang FJ (2008) Intelligent reservoir operation system based on evolving artificial neural networks. *Adv Water Resour* 31:926–936. <https://doi.org/10.1016/j.advwatres.2008.03.002>
- Chaves P, Tsukatani T, Kojiri T (2004) Operation of storage reservoir for water quality by using optimization and artificial intelligence techniques. *Math Computers Simul* 67(4–5):419–432. <https://doi.org/10.1016/j.matcom.2004.06.005>
- Danilov D, Zhigljavsky A (1997) Principal components of time series: the caterpillar method. University of St Petersburg, St. Petersburg (**In Russian**)
- Darlane AB, and Karami F (2012) Comparison of hedging policies in reservoir management under drought condition. In: 10th International congress on advances in civil engineering, pp. 17–19 October, Middle East Technical University, Ankara, Turkey
- Darlane AB, Karami F (2014) Deriving hedging rules of multi-reservoir system by online evolving neural networks. *Water Resour Manage* 28(11):3651–3665. <https://doi.org/10.1007/s11269-014-0693-0>
- Darlane AB, Moradi AM (2014) A comparative analysis of evolving artificial neural network and reinforcement learning in stochastic optimization of multireservoir systems. *Hydrol Sci J*. <https://doi.org/10.1080/02626667.2014.986485>
- Fesanghary M, MahdaviMinary-Jolandan MM, Alizadeh Y (2008) Hybridizing harmony search algorithm with sequential quadratic programming for engineering optimization problems. *Comput Methods Appl Mech Eng* 197(33–40):3080–3091. <https://doi.org/10.1016/j.cma.2008.02.006>
- Geem ZW, Kim JH, Loganathan GV (2001) A new heuristic optimization algorithm: harmony search, society for modeling and simulation international (SCS). *SIMULATION* 76(2):60–68. <https://doi.org/10.1177/003754970107600201>
- Ghil M, Taricco C (1997) Advanced spectral analysis methods. In: Castagnoli GC, Provenzale A (eds) Past and present variability of the solar-terrestrial system: measurement”, data analysis and theoretical models. IOS Press, Amsterdam
- Golyandina N, Nekrutkin V, Zhigljavsky A (2001) Analysis of time series structure: SSA and related techniques. Chapman and Hall/CRC, London
- Hasebe M, Nagayama Y (2002) Reservoir operation using the neural network and fuzzy systems for dam control and operation support. *Adv Eng Softw* 33:245–260. [https://doi.org/10.1016/S0965-9978\(02\)00015-7](https://doi.org/10.1016/S0965-9978(02)00015-7)
- Hassani H, Mahmoudvand R (2013) Multivariate singular spectrum analysis: a general view and new vector forecasting approach. *Int J Energy Stat* 1(1):55–83. <https://doi.org/10.1142/S2335680413500051>
- Jain SK, Das A, Srivastava DK (1999) Application of ANN for reservoir inflow prediction and operation. *J Water Resour Plan Manag* 125(5):263–271. [https://doi.org/10.1061/\(ASCE\)0733-9496](https://doi.org/10.1061/(ASCE)0733-9496)
- Lee KS, Geem ZW (2005) A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. *Comput Methods Appl Mech Eng* 194:3902–3933
- Mahdavi M, Fesanghary M, Damangir E (2007) An improved harmony search algorithm for solving optimization problems. *Appl Math Comput* 188:1567–1579. <https://doi.org/10.1016/j.amc.2006.11.033>
- Marques CAF, Ferreira JA, Rocha A, Castanheira JM, Melo-Goncalves P, Vaz N, Dias JM (2006) Singular spectrum analysis and forecasting of hydrological time series. *Phys Chem Earth* 31:1172–1179. <https://doi.org/10.1016/j.pce.2006.02.061>
- Myung NK. (2009) Singular Spectrum Analysis, University of California, Los Angeles, Master of Science thesis
- Omran GH, Mahdavi M (2008) Global-best harmony search. *Appl Math Comput* 198:643–656. <https://doi.org/10.1016/j.amc.2007.09.004>
- Pan QK, Suganthan PN, Tasgetiren MF (2009) A harmony search algorithm with ensemble of parameter sets. *Evol Comput*. <https://doi.org/10.1109/CEC.2009.4983161>
- Pan QK, Suganthan PN, Tasgetiren MF, Liang JJ (2010) A self-adaptive global best harmony search algorithm for continuous optimization problems. *Appl Math Comput* 216(3):830–848. <https://doi.org/10.1016/j.amc.2010.01.088>
- Pianosi F, Thi XO, Soncini-Sessa R (2011) Artificial neural networks and multi objective genetic algorithms for water resources management: an application to the Hoabinh reservoir in Vietnam. In: Proceedings of the 18th IFAC World Congress, Milano (Italy), August 28 - September 2, <https://doi.org/10.3182/20110828-6-IT-1002.02208>
- Sivapragasam C, Liong S-Y, Pasha MFK (2001) Rainfall and runoff forecasting with SSA–SVM approach. *J Hydro inform* 3:141–152
- Wu CL, Chau KW (2011) Rainfall–runoff modeling using artificial neural network coupled with singular spectrum analysis. *J Hydrol* 399:394–409. <https://doi.org/10.1016/j.jhydrol.2011.01.017>
- Wu CL, Chau KW, Li YS (2009) Predicting monthly streamflow using data-driven models coupled with data-preprocessing techniques. *Water Resour Res* 45:W08432. <https://doi.org/10.1029/2007W0006737>
- Wu CL, Chau KW, Fan C (2010) Prediction of rainfall time series using modular artificial neural networks coupled with data-preprocessing techniques. *J Hydrol* 389:146–167. <https://doi.org/10.1016/j.jhydrol.2010.05.040>
- Yiou P, Sornette D, Ghil M (2000) Data-adaptive wavelets and multiscale SSA. *J Phys D* 142:254–290. [https://doi.org/10.1016/S0167-2789\(00\)00045-2](https://doi.org/10.1016/S0167-2789(00)00045-2)