# Short-term demand forecast using a bank of neural network models trained using genetic algorithms for the optimal management of drinking water networks

Hector Rodriguez Rangel, Vicenç Puig, Rodrigo Lopez Farias
and Juan J. Flores

## ABSTRACT

Efficient management of a drinking water network reduces the economic costs related to water production and transport (pumping). Model predictive control (MPC) is nowadays a quite well-accepted approach for the efficient management of the water networks because it allows formulating the control problem in terms of the optimization of the economic costs. Therefore, short-term forecasts are a key issue in the performance of MPC applied to water distribution networks. However, the short-term horizon demand forecast in a horizon of 24 hours in an hourly based scale presents some challenges as the water consumption can change from one day to another, according to certain patterns of behavior (e.g., holidays and business days). This paper focuses on the problem of forecasting water demand for the next 24 hours. In this work, we propose to use a bank of models instead of a single model. Each model is designed for forecasting one particular hour. Hourly models use artificial neural networks. The architecture design and the training process are performed using genetic algorithms. The proposed approach is assessed using demand data from the Barcelona water network.

Key words | artificial neural networks, demand forecast, genetic algorithms, model predictive control, water distribution networks

**Hector Rodriguez Rangel** (corresponding author)
Instituto Tecnológico de Culiacán, División de Estudios de Posgrado e Investigación,
Juan de Dios Bátiz 310 pte. Col. Guadalupe C.P., 80220 Culiacán,
Mexico
E-mail: *hrodriguez@itculiacan.edu.mx*

**Vicenç Puig**
Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Carrer LLorens Artigas, 4-6,
08028 Barcelona,
Spain

**Rodrigo Lopez Farias**
Centro de Cómputo y Procesos de Información Universitaria,
Universidad Michoacana de San Nicolás de Hidalgo, Morelia,
Mexico

**Juan J. Flores**
División de Estudios de Postgrado, Facultad de Ingeniería Eléctrica,
Universidad Michoacana de San Nicolás de Hidalgo, Morelia,
Mexico

## INTRODUCTION

Drinking water is one of our vital resources; it is used in almost every human activity (personal hygiene, cleaning in general, manufacturing products, etc.). In fact, according to Primary Health Care in Alma-Ata in 1978, safe water is a main component for the World Health Organization (1993).

Usually, drinking water is brought from distant sites to cities and its production and transportation is expensive. Water is stored in temporary tanks and then throughout the city. Water distribution systems consist of an interconnected series of pipes, storage facilities, and components that convey drinking water to meet the city water demand (Alperovits & Shamir 1977). The economic costs associated with drinking water production are due to chemicals, legal canons, and electricity costs. Moreover, the transportation of drinking water through the overall water network plays an important role regarding electricity costs in pumping stations. Improving water supply management means a reduction in operating costs, avoiding the development of new supplies and unnecessary expansion of infrastructure (Stephenson 1998). It also reduces withdrawals from limited freshwater supplies, reducing at the same time the negative effect on the natural environment.

There are several strategies to manage the water supply network efficiently. One of them is model predictive control (MPC), an optimization-based control strategy applicable to

a wide range of industrial applications (Ormsbee & Lansey 1994; van Overloop 2006; Ma *et al.* 2012). MPC provides suitable techniques to compute optimal control strategies ahead of time for all the control elements of a water system supply. The accuracy of MPC depends on the water distribution model and the accuracy of the short-term forecasting of the water demand. In water distribution networks, MPC focuses on finding the best input control sequence for controlling the water supply network for the next 24 hours. A poor demand prediction may lead to a significant increase in costs (for example, in Hippert *et al.* (2001), it is stated that an increase of 1% in the error would imply a £10 million increase of operational cost).

In this paper, we address the short-term water demand forecasting problem, in particular, the water demand forecasting for the next 24 steps ahead (hourly prediction). To deal with this problem, we propose the fragmentation of the forecasting model. Instead of having a unique forecasting model of the next 24 hours, we explore the use of 24 models. Each model is specialized in one particular hour in the 24 hour horizon and is independent from the other 23 models. That is, Model 1 provides the forecast for the next hour $yt + 1$, Model 2 provides the forecast for the 2 hours ahead $yt + 2$, and so on, until Model 24 that provides the forecast for the 24 hours ahead $yt + 24$. The idea of using multimodels came from other methods of solution, such as solving higher order systems, reducing them to their simplest form (e.g., linearization of nonlinear systems by Seron & Braslavsky (2000)). Also, if we compare a one-step optimization function against multiple-steps ahead, it is clearly that the one-step ahead is simpler than the case of multiple steps; and therefore, the threshold of uncertainty and computational time grows as the number of steps (McElroy & Wildi 2013).

Since each model is independent, it is necessary to create, define, and train 24 models. Recent research activities on artificial neural networks (ANNs) have shown that they have powerful pattern classification and recognition capabilities. Inspired by biological systems, particularly by research into the human brain, ANNs are able to learn and generalize from experience. Currently, ANNs are used for a wide variety of tasks in many different fields of business, industry, and science (Zhang *et al.* 1998).

One major application area of ANNs is forecasting. ANNs provide an attractive alternative tool for both forecasting researchers and practitioners. Several distinguishing features of ANNs make them valuable and attractive for a forecasting task (Zhang *et al.* 1998).

Given a time series, we need to provide a neural model capable of producing an acceptable forecasting of the process it represents. We explore the use of ANNs to define each model. The use of an ANN as a forecasting method has shown excellent performance in engineering applications. However, one of the drawbacks resides in the difficulty of training the ANN model. The ANN optimization modeling problems often lead to non-convex optimization problems. This fact implies that global optima cannot be guaranteed. Therefore standard gradient-based algorithms can only guarantee suboptimal results. Hence, the exploration of the gradient-free optimization algorithms (such as genetic algorithm (GA)) is considered as a suitable training method (Nocedal & Wright 1999).

The more information the ANN received (including both qualitative and quantitative data), the more accurate the output will be. A time series is a sequence of chronologically ordered observations recorded at regular time intervals. Water demand time series present different dynamic behaviors that might be seen as the change of different possible unknown regimes. For example, the water demand behavior during the holidays is different from that of working days, or summer from winter days, but regular during these periods. This fact motivates considering these behaviors by adding the prediction of next qualitative behavior (mode) as an input to the ANN.

In summary, we present a methodology that fragments the simultaneous demand forecasting of the 24 hours ahead into smaller subproblems (models), where each model is independent from each other. The forecast is not only a function of past observations; an external factor, social behavior (called a pattern) is added. To estimate the next pattern, first we detect the patterns in the time series, then we label each day according to its pattern, to finally be able to estimate the next pattern. The forecasting method used in each model is based on an ANN. The inputs of the ANN are the past observations plus the estimation of the next day pattern. We design the architecture of the net (number of inputs, hidden layers) using GAs; traditionally, this type of process is carried out by trial and error. For the learning process, we use GA and compare

them with traditional methods (e.g., back propagation (BP), Broyden–Fletcher–Goldfarb–Shanno (BFGS), etc.). GA exhibited the best performance. Finally, we provide the worst-case forecasting scenario by implementing the confidence interval of each model, yielding a boundary of the prediction. This worst-case scenario is important for the MPC controllers to prevent a possible lack of the drinking water resource.

After having presented the proposed approach, the rest of the paper is organized as follows. 'Related work' surveys the state of the art in general forecasting methods applied to water demand. Next, 'Case study' describes the water distribution system of Barcelona city. The following section proposes the use of a set of models to forecast the next 24 hours. The final two sections discuss the results obtained and the conclusions drawn.

## RELATED WORK

In the literature regarding time series (i.e., Hamilton 1994; William & Shyong 1994), there is a strong effort to find the best way to decompose time series into several simpler time series to fit better models that improve the forecasting performance. This is not an easy task since in many real cases there is not an available model that describes the dynamic fluctuation of the data. In the early successful stage in statistics, the *divide and conquer strategy* has been used; for example, the decomposition into different basic components that might explain the general dynamics of time series, such as trend, seasonal, random, and cyclical components that are integrated in the autoregressive integrated moving average (ARIMA) methodology (Harvey 1990; Contreras *et al.* 2003).

Taieb *et al.* (2010) perform a study of multiple steps ahead of forecasting. They compare the *iterated* and *direct* forecasting techniques. The iterated method uses a one-step ahead predictor; once the predictor estimates the future series value, this value is re-injected as an input to the next prediction. The direct methods estimate the next *H* steps, by the prediction of a set of *H* predictive models. In their research, they conclude that direct methods often require higher functional complexity than iterated ones, but direct models present a better accuracy than iterated ones.

Sorjamaa *et al.* (2007) used direct forecast, using *H* models to define the next *H* steps ahead. This work is similar to the methodology presented in Sorjamaa's study, but differs in the way of modeling the time series and selecting entries for the forecast. While we use ANN with GA, they used least square support vector machines and k-nearest neighbors (kNN) to discriminate certain entries.

Nowadays, with the growth of computational resources and the development of machine learning and pattern recognition algorithms (i.e., Olszewski 2001; Liao 2005, 2007), it is possible to analyze more complex time series. There are practical cases where the single linear modeling approach is not enough for systems that present different behaviors along time (Martinez Alvarez *et al.* 2007; Kumar & Patel 2010; Benmouiza & Cheknane 2013; Quevedo *et al.* 2014). These behavior changes might be produced by changes of dynamical regimes. Although the multi-modeling approach was born with the analysis of partially known real systems, the same ideas can be adopted for time series forecasting, where there is no knowledge about the system behind the dynamics, as is the case of water demand time series.

In relation to water demand, different model methodologies for forecasting have been explored (Donkor *et al.* 2012). We found in the literature modeling methodologies (i.e., based on Box–Jenkins, ANNs, Holt–Winters, ARIMA, wavelets, etc.) that deal with this problem (Zhou *et al.* 2000; Alvisi *et al.* 2007; Al-Hafid & Al-maamary 2012; Quevedo *et al.* 2014; Tiwari & Adamowski 2015). Also, Melgoza *et al.* (2004) proposed a technique for prediction of electrical demand based on multiple models; each model describes a region of behavior of the system, called operation regime.

Adaptive oriented predictive methods are also found in the literature, e.g., the algorithm proposed by Bakker *et al.* (2013), which considers the last 2 days for predicting the water demand of the next 2 days. The contribution of the days is weighted and a fixed calendar is considered as an additional input. After tuning the day weights, it derives day factors and daily demand patterns weekly.

This paper is related to research in the area of ANN for the task of time series forecasting in the works of Bakirtzis *et al.* (1996); Jain *et al.* (2001), and Liu *et al.* (2003), but differs in the way of creating the architecture of the ANN, and also the way to train it. In previous works, Flores *et al.* (2009, 2010, 2012), have used evolutionary algorithms (EA) to define the

architecture of the ANN and train it; in these previous works the forecast accuracy showed a better performance than traditional gradient-based training algorithms.

One of the inputs of the ANN is the mode (or regime) of the next day to forecast. In previous works related to regimes, the identification of regime behavior has used qualitative information. Benmouiza & Cheknane (2013) proposed the implementation of a global non-linear autoregressive (NAR) neural network predictor to estimate the regimes associated with another local NAR neural network predictor for the hourly global solar radiation. Kumar & Patel (2010) proposed a predictive algorithm using data clustering and local training models that combined produce the forecast. Martinez Alvarez et al. (2007) used clustering to group the days with similar patterns. Regarding the variation of the water demand in working days and holidays, Quevedo et al. (2014) developed a daily seasonal ARIMA model combined with an hourly pattern. In their approach, seasonal ARIMA predicts total days' consumption and a daily pattern is selected according to a calendar for distributing the hourly consumption of the day.

The work of Cutore et al. (2008) is similar to our approach. It uses an ANN as prediction model, and uses human behavior (modes) as an input of the ANN, but the modes are extracted from a calendar. Moreover, a network training procedure is considered through the Shuffled Complex Evolution Metropolis algorithm. In their work they also consider climate variables, but these last ones, did not improve the prediction results.

Also, our work is similar to the work of Romano & Kapelan (2014), since both works split the forecast into 24 models, and use ANN as the forecast method, but they differ in the kind of inputs and are completely different in the learning process. Regarding the selection of inputs, Romano & Kapelan (2014) use a lag window of past demand data, the day of the week, and the hour associated with the forecast horizon. The day of the week and the hour are used to associate the water demand time series with human behavior patterns. In the literature, we found works where these human patterns are incorporated into the forecast methodology (e.g., Quevedo et al. 2010), but each work differs in the way they are incorporated. In our work, an intelligent method to estimate the next day pattern is used. This method starts pre-processing the time series by

using the silhouette coefficient to define the number of patterns (or modes). Each day is labeled in the training set according to its pattern, and we estimate the next day pattern using kNN in the validation set. We would like to mention that in our previous works (Quevedo et al. 2010; Lopez et al. 2015a), we started exploring the use of a calendar (days of the week, in terms of labor days and holidays), but we found that this intelligent method improves the calendar method (Lopez et al. 2015b). Regarding the learning process, it is performed in a completely different way. Romano & Kapelan (2014) used traditional gradient-based learning algorithms (e.g., BP, conjugate gradient, Levenberg–Marquardt), while we use GA to perform the learning process. In our work, we perform a comparison with a single ANN performing one-step ahead forecasting, using for the learning process: traditional methods (BP, BFGS, conjugate gradient algorithm), and GAs. GAs outperformed the other methods, with respect to forecast accuracy. Proofs of these experiments are provided later in the paper. In summary, the proposed approach improves the use of the calendar (day of the week) by using the estimation of the next day pattern. Also, it improves the accuracy obtained by using GA compared to the use of traditional gradient-based learning methods (e.g., BP).

## CASE STUDY

The Barcelona water network will be the case study used to illustrate the proposed methodology. The Barcelona network is managed by the company Aguas de Barcelona. This company not only supplies water to Barcelona city but also to the metropolitan area. The network supplies 23 municipalities in a 424 km$^2$ area with 4,645 km of pipes in order to meet the water demands of about three million people (Sociedad General de Aguas de Barcelona 2008).

The sources of water are the rivers Ter and Llobregat that are regulated at the head by some dams with an overall capacity of 600 hm$^3$. There are four drinking water treatment plants which treat the underground flows. Also, there are several underground sources (wells) that can provide water through pumping stations.

The different water sources currently provide a flow of around 7 m$^3$/s. Currently, a desalter plant has just been

put into production with a capacity of 60 hm$^3$/year. This plant is located at the end of the Llobregat river and produces drinking water by treating the sea water through a desalinization process. This plant will become of prime importance especially in drought periods since it will help maintain the water supply.

Due to the geographical topology of Barcelona and its surroundings, the water network supply of the metropolitan area is structured in pressure zones. Indeed, the topology of Barcelona, with a big difference in height between the sea level and the highest point to be supplied which is about 500 m above the sea level, makes it necessary to homogenize the pressure by intermediate tanks and pumping stations.

The complete transport network is composed of 63 storage tanks, three surface sources and seven underground sources, 79 pumps, 50 valves, 18 nodes, and 88 demand sectors (see Figure 1). The network is controlled through a

SCADA system with sampling period of 1 hour. For the MPC control scheme, a prediction horizon of 24 h is chosen for the model and demand forecasts. These forecasts are updated every hour with the recently updated measurements following the receding horizon philosophy used in MPC, as discussed in the 'Introduction'.

This case study has already been considered for demand forecasting by Quevedo et al. (2014), with several time series approaches (ARIMA, Holt–Winters, and BSM) being used for the same purpose as the approach proposed in this paper. The same case study and methods have also been used for the validation and reconstruction of flow meter data. Differently from these previous approaches, here a bank of ANN models are trained using GA and used for forecasting the demand in a 24 hour horizon with a time scale of 1 hour in a way that can be easily integrated with the MPC controller of the water network.
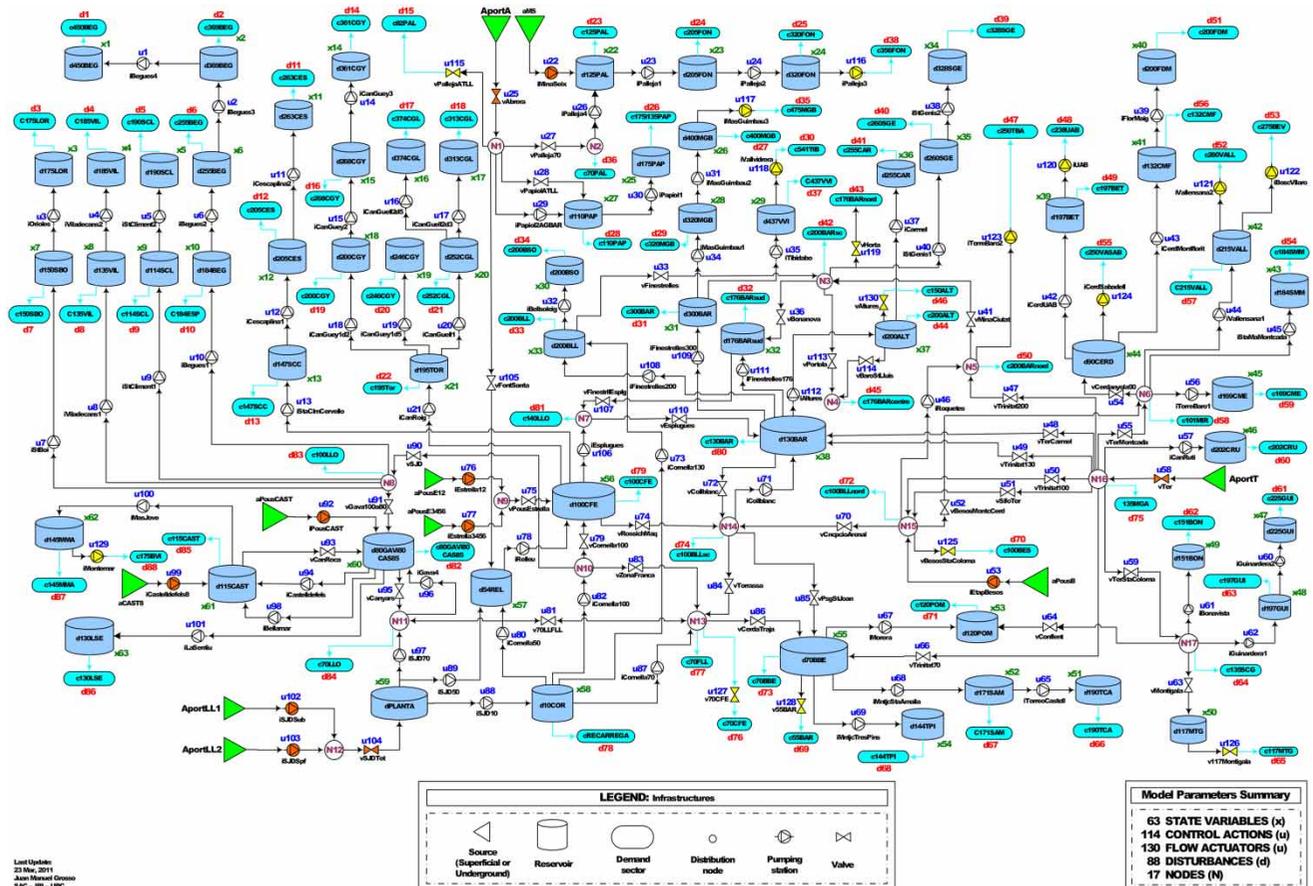


**Figure 1** | Barcelona water network.

## PROPOSED APPROACH

Water consumption for operational control of water networks in urban areas is usually managed on a daily basis, because reasonably good hourly 24 hour ahead demand predictions may, in general, be available, and common transport delay times between the supplies and the consumer sites allow operators to follow daily water request patterns. Therefore, this horizon is appropriate for evaluating the effects of different control strategies on the water network, with respect to operational goals. However, other horizons may be more appropriate in specific utilities. The short-term forecast of the intraday series has a main feature: the double periodicity (daily and hourly). Moreover, the consumption of water demand changes between holidays and working days and between summer and winter, but present regular behaviors during these periods. Each behavior can be represented by a regime that is characterized by a mode (qualitative behavior). Figure 2(a) shows 1 month of qualitative behavior where the time series is segmented; each line of the plot corresponds to 1 day. From Figure 2(a), we can observe two qualitative patterns. In Figure 2(b), two unitary prototypes or descriptors are plotted representing the two main kinds of behaviors.

In the literature, we found a work that indicates the conditions under which a multi input–multi output (MIMO) ANN architecture is less accurate than a multi input–single output (MISO) architecture (Chang *et al.* 2007). This is because the forecasting problem using MIMO architectures becomes a multi-objective optimization problem. Therefore, to address the water demand forecasting problem, we use MISO models.

In this work, we translate the problem of one MIMO model into 24 single independent MISO models. We propose the use of a bank of 24 models, where each model is independent and focuses specifically on the prediction of 1 hour (for example, Model 1 is specialized in the prediction of $y_{t+1}$, Model 2 in $y_{t+2}$, and so on). Figure 3 shows that the problem is split into different MISO models, where each model receives multiple inputs ($k$ past observations, plus the next qualitative pattern behavior of the water consumption) and one single output.

Each model receives as input the past observations of the time series and the estimation of the next mode. The independence of each model means that each model may be defined arbitrarily (for example, one model may be defined using ANN, another using Holt–Winters while the others using ARIMA). Also, each model defines the number of previous observations (that is the structure) to use in the prediction. In our case, each model in the bank of models (BM) is defined by a three-layer fully connected ANN while the architecture and the training of the net is performed by GAs. The inputs and the outputs of each model are defined in Table 1.

Table 1 shows that every model is used to predict a particular step in the 24-step horizon; e.g., Model 1 defines $y_{t+1}$,
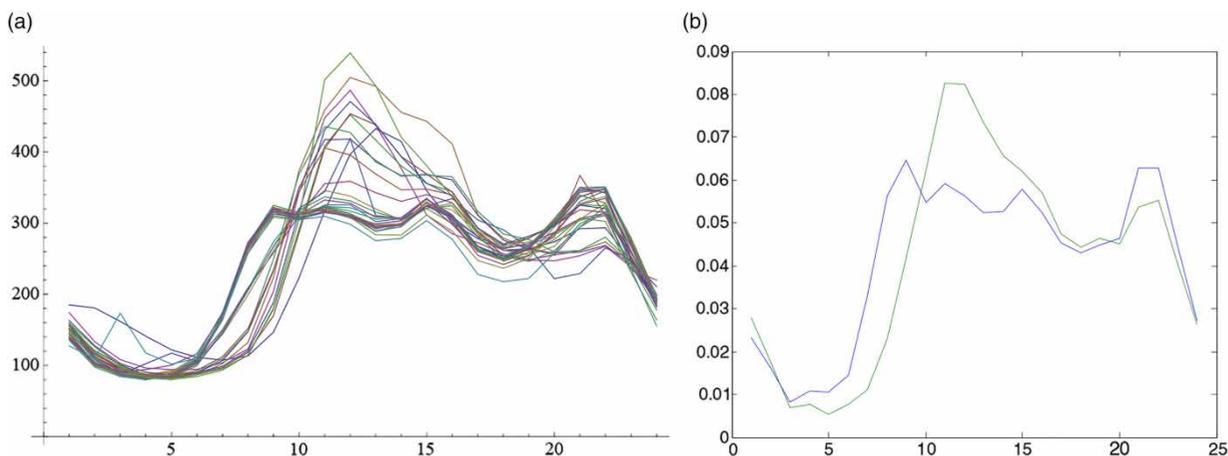


**Figure 2** │ Qualitative behavior of the water demand. From (a) we segment the time series per days; in (b) we observe the two different behaviors (patterns) of (a). (a) Daily distribution water demand patterns of p10012. (b) Daily distribution water demand patterns prototypes of p10017.
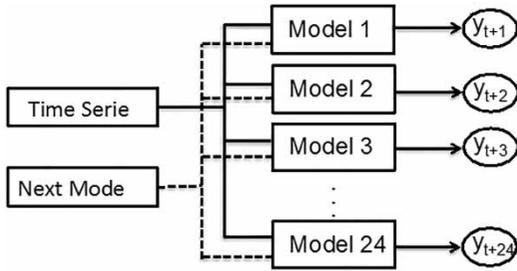
**Figure 3** │ Fragmentation of the forecasting into a BM (24 models) where each model corresponds to a different forecasting step.

**Table 1** │ Inputs and outputs of each model defined in the BM

|  | Input | Output |
|---|---|---|
| Model 1 | $\{y_{t-k1}, y_{t-k1+1}, y_{t-k1+2}, \ldots, y_{t-1}, y_t, M_i\}$ | $y_{t+1}$ |
| Model 2 | $\{y_{t-k2}, y_{t-k2+1}, y_{t-k2+2}, \ldots, y_{t-1}, y_t, M_i\}$ | $y_{t+2}$ |
| Model 3 | $\{y_{t-k3}, y_{t-k3+1}, y_{t-k3+2}, \ldots, y_{t-1}, y_t, M_i\}$ | $y_{t+3}$ |
| $\ldots$ | | |
| Model 23 | $\{y_{t-k23}, y_{t-k23+1}, y_{t-k23+2}, \ldots, y_{t-1}, y_t, M_i\}$ | $y_{t+23}$ |
| Model 24 | $\{y_{t-k24}, y_{t-k24+1}, y_{t-k24+2}, \ldots, y_{t-1}, y_t, M_i\}$ | $y_{t+24}$ |

where $y_t$ is the observation at the time $t$, and $y_{t+i}$ is the next observation that is going to be forecasted. Also, every model receives a different number of inputs because the GA defines for every case (model) the number of last observations ($k$) to consider (e.g., Model 1 has $k_1$ inputs).

In this paper, the ANN architecture used for this task is a multi-layer perceptron (MLP). A MLP, as a universal approximator, can learn any given function, as long as it has enough neurons in the hidden layer. That fact allows the network to capture the different forms of the function to be modeled. An example of this kind of architecture is depicted in Figure 4.

Figure 4 graphically observes the proposed network. It has $k + 1$ inputs, which are selected from the $k$ previous measurements, plus the next mode. The next mode is the estimation of the next day qualitative behavior (i.e., labor day or holiday). The process to estimate the next mode is performed as described in the work of Lopez *et al.* (2015b). The hidden layer has $m$ neurons, the ANN has one output (the corresponding predicted hour ($\hat{y}_{t+i}$)), and the sigmoid function as the activation function. In summary, the forecast
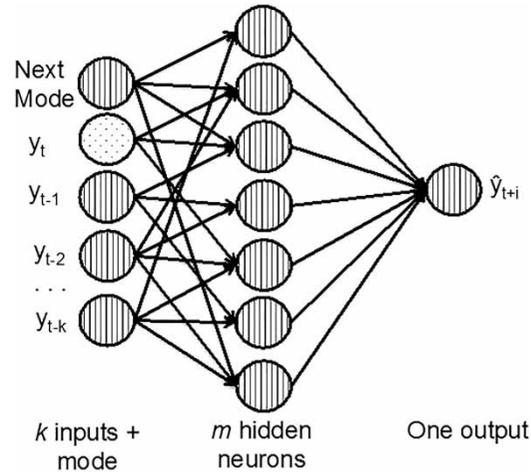


**Figure 4** │ MLP architecture with $k$ inputs plus the next mode, $m$ hidden neurons in the hidden layer and one output ($\hat{y}_{t+i}$).

function to $\hat{y}_{t+i}$ is defined as:

$$\hat{y}_{t+1} = f_1\left(\sum_{l}^{k+1} w_l x_l\right)$$
$$x_l = f_2\left(\sum_{j}^{m} w_{lj} y_{t-lj}\right) \tag{1}$$

where $f_1$ and $f_2$ are the activation functions, and $w$ are the coefficients (also known connection weights).

In this forecast problem, we need to provide an accurate model that defines the behavior of the water demand. To define the accuracy of each model, we use the minimization of the mean square error statistical measure, which is defined as:

$$MSE = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \tag{2}$$

To obtain an accurate $\hat{y}$, it is necessary to define the number of inputs (past observations $k$), number of neurons in the hidden layer ($m$), and the connection weights ($w_l$) of each ANN defined in the BM (for example, for one net of 25 inputs and 50 neurons in the hidden layer, we have the problem of determining more than 1,000 connection weights ($w_l$)). Thus, the determination of the optimal number of neurons in the hidden layer is a crucial issue. If the hidden layer is too small, the network cannot possess

sufficient information processing power, and thus yields inaccurate forecasting results. On the other hand, if it is too large, the training process will be very long. The optimization problem (2) is a non-convex problem, where it is hard to reach a global solution, especially when using gradient-based algorithms, because it depends on the initial conditions. Therefore, for the training process, we propose the use of evolutionary computation (specifically GA), already introduced in Flores *et al.* (2009, 2010, 2012), showing better performance than traditional methods (*gradient descent*) according to the comparisons performed.

## Training the ANN with GA

GA is an optimization technique inspired by Darwin's principle of evolution. That is, it mimics a simplistic version of the process of biological evolution, which consists of creating a population of individuals, where each individual represents a prospective solution of the problem being solved. GA modifies this population using genetic operators: selection, mutation, recombination, etc. This stage, called a generation, repeats until a termination criterion is met. At the end of the process, the best individual (i.e., the fittest one) found during the evolution is returned as the solution of the problem.

Determining the best ANN architecture for forecasting is an optimization problem. In this paper, we use GA to find the optimal ANN architecture and its connection weights. GA defines the architecture of the ANN and the weights of the neurons' connections. Each individual (the chromosome) is defined as a vector of real numbers, where the first two values of the vector define the architecture of the net. The first position defines the number of previous measurements to use as an input, and the second position defines the number of neurons in the hidden layer. The following data are two sets of reals, corresponding to the weight connections. The first set corresponds to the weight connections from the input to the neurons in the hidden layer; its size is $(k+1)*m$ ($k$ is the number of inputs, and $m$ is the number of neurons in the hidden layer). The second set corresponds to the connection weights from the neurons in the hidden layer to the output neuron; its size is $m$. Figure 5 shows the structure of the chromosome.
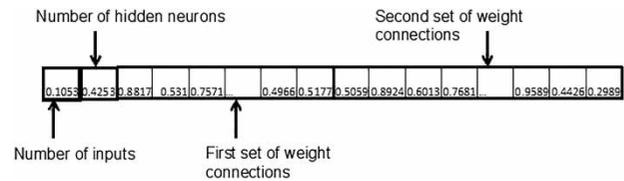


**Figure 5** | The chromosome of the ANN represented as a vector of reals. The first two positions of the vector define the structure of the net, and the following data correspond to the connection weights.

The chromosome vector is coded in values between zero and one. To decode the vector, we use

$$DV = ((LimMax - LimMin)*realData + LimMin) \qquad (3)$$

where *LimMax* is the upper limit and *LimMin* is the lower limit to decode. For instance, if we want to test architectures between 10 and 80 inputs, the upper limit and the lower limit will be 80 and 10 respectively; and DV is the decoded value. Notice that Equation (3) is based on a maximum and a minimum value. This is because the first two positions of the vector define the number of inputs and number of neurons in the hidden layer (e.g., values between 10 and 80). The next part of the vector corresponds to the weights of the connections (with values between −1 and 1).

The regimes (or modes) are determined as defined in the work of Lopez *et al.* (2015b). In this work, the regimes of a water demand time series are obtained using k-means and validated with the silhouette coefficient to select the number of clusters. Once we have labeled every day with a mode, we estimate the next mode using kNN.

The forecast is defined in two processes: the offline process and the online process. The offline process defines the modes and the 24 MISO models, forming the BM. In the online process, we estimate the next mode and the next day water demand (hourly periods) from the ANNs already trained in the offline process.

## Offline process

To perform a forecast, first we require to segment the time series in prototypes or modes (see the work of Lopez *et al.* (2015b)). Then, we need to define the architecture and the connection weights of the 24 ANNs.

The process to define the ANNs in the BM is defined in Algorithm 1. From this algorithm, we start defining the

number of prototypes or modes of the time series (*Define-TimeSeriePrototypes*). Once the prototypes are obtained, we label each day of the time series according to its prototypes. This step is performed according to a previous work (Lopez *et al.* 2015b). The function *LabelTimeSeries* receives as an input the time series and the prototypes defined before. Using the function, we label each day according to its pattern and return the vector of labels (or modes).

With the modes vector and the time series, we start defining each ANN contained in the BM. The parameters to define each ANN are the number of past observations, number of hidden neurons, and the connection weights. The function *TunningANNparameters* optimizes these parameters, with the consideration of each net specialized in forecasting only the *ModelNumber* step ahead. This function returns a coded vector (as in Figure 5) of reals that define each ANN.

**Algorithm 1** Forecast 24 steps ahead (*TimeSeries*)

1: *Prototypes* ← DefineTimeSeriesPrototypes(*TimeSeries*)
2: *Modes* ← LabelTimeSeries(*TimeSeries, Prototypes*)
3: **for** $i \leftarrow 1$, $i <= 24$ **do**
4: *ModelNumber* ← $i$
5: *ANNModel* ← TuningANNparameters(*TimeSeries,*
6: *Modes, ModelNumber*)
7: *BM* ← InsertInToBankOfModels(*ModelNumber, ANNModel*)
8: **end for**
9: return *BM*

Algorithm 2 defines the architecture of the ANN and performs the learning process based on GA. It starts by generating a random population by means of the function *GenerateInitialPopulation*. Each individual in the population is encoded using real numbers between 0 and 1 according to Figure 5. To decode the chromosome we use (3). The next step is to evaluate each individual of the population. The evaluation is performed according to the optimization function (2).

**Algorithm 2** Tuning ANN parameters (*TimeSeries, Modes, ModelNumber*)

1: *pop* ← GenerateInitialPopulation()
2: *pop* ← EvaluateIndividuals(*pop, ModelNumber*)
3: **while** NOT Finish **do**
4: **for** j ← 0, *j < Length(pop)* **do**
5: *Parents* ← SelectParents(*pop*)
6: *offspring* ← Crossover(*Parents*)
7: *offspring* ← Mutation(*offspring*)
8: *offspring* ← EvaluateIndividuals(*offspring, ModelNumber*)
9: *offspringList* ← InsertInToDescendentList(*offspring*)
10: **end for**
11: **if** Convergence Criteria Achieved **then**
12: *Finish* ← TRUE
13: **end if**
14: **end while**
15: **return** *BestIndividual*

After the evaluation of each individual the evolution process is started, the parents are selected and, in our case, we use tournament selection. With a certain probability we perform the crossover and the mutation giving as a result two offspring. The offspring is evaluated and inserted in the offspring list. This process is repeated until a convergence criterion is achieved. In our case, the search is limited to a predefined number of evolutions.

# RESULTS

To provide an empirical proof of the effectiveness of our approach, real water demand coming from the Barcelona case study presented earlier is used. In particular, demand time series obtained from flow meters p10017, p10012, p10025, and p10031 are used. The experiments were divided in two phases: the *offline* and the *online*. In the *offline* process, we define the number of modes or patterns of the time series; the time series per day is labeled according to its pattern number, forming the mode vector. The mode of the next day is used as an input of each model.

Once the mode vector is obtained, we split the problem to produce 24 ANN models, where every ANN model forecasts only one of the 24 hours. For this part, we use Algorithm 1. GAs are used in Algorithm 2 to automatically define the architecture of the net, and perform the training process. After performing the *offline* process, we perform the forecast (*online* phase) by using the models already trained. All the experiments were performed on the Python platform (Python Software Foundation 2016), using the GA Library of the package *Distributed Evolutionary Algorithms* (Fortin *et al.* 2012).

As mentioned before, we start performing the experiments with traditional methods (gradient-based methods defined in the *Neurolabs Library*). Table 2 shows a comparison between the traditional methods defined in the *Neurolabs Library* and GA. The experiments presented in this table were performed using a water demand time series of the flow meter p10012. From Table 2, we observe that GA presents a better performance in the validation set than the other three methods.

The following experiments were performed in a 30–60 day period (which give us a time series of 720–1,440 measurements). The training and validation sets were created using 70% and 30% of these data, respectively. After the training process, we obtained 24 trained nets. The GA parameters used to perform the training process are shown in Table 3. The inputs of the ANN contain between 10 and 70 past observations. This means that GA will search within this range; the same case applies for the number of neurons. In the case of the mutation probability, it means that 55% of the offspring will mutate, but only less than 1% of its genes are mutated. The approximate time to train a single net is about 90 minutes.

Once the parameters (number of inputs, hidden neurons, weights connections) of the 24 models in the BM have been tuned, we perform the forecast with the

**Table 2** │ Comparison between traditional methods versus GA

| Method | MSE training | MSE validation |
| --- | --- | --- |
| Gradient descent backpropagation | 0.42883913 | 0.42747427 |
| BFGS | 0.0000124 | 0.00150682 |
| Conjugate gradient algorithm | 0.00159929 | 0.00292422 |
| GAs | 0.001002 | 0.0010163 |

**Table 3** │ GA parameters for ANN optimization

| Parameter | |
| --- | --- |
| Number of inputs | 10–70 |
| Number of hidden neurons | 20–70 |
| Initial population | 250 |
| Number of evolutions | 350 |
| Crossover probability | 70% |
| Chromosome mutation probability | 55% |
| Gene mutation probability | 0.5% |

validation set. Figure 6 illustrates the performance of 1 day forecasting of the four considered demands. Each hour is defined by one model in the set of models.

During the training process, the MSE function (2) is minimized. After the training (offline) process and to assess and compare the performance of the proposed approach different statistical measures (mean absolute error (MAE), mean absolute percent error (MAPE), and root mean square error (RSME)) were considered. Definitions of these statistical measures are:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |\hat{y}_i - y_i| \qquad (4)$$

$$MAPE = \frac{100}{N} \sum_{i=1}^{N} \frac{|\hat{y}_i - y_i|}{\bar{y}} \qquad (5)$$

$$RMSE = \sqrt{MSE} \qquad (6)$$

Figure 6 illustrates the performance of 1 day forecasting of the four flow meters; the performance across the whole validation set is shown in Figure 7. Figure 7 shows an hour by hour comparison of the whole validation set in terms of the square (MSE), absolute (MAE), and relative percentage (MAPE) errors. The figure contains three lines, each one corresponds to the error magnitude in terms of MAE, MAPE, and MSE of the flow meter p10017.

The proposed approach is compared against standard methods (one single ANN, Holt–Winters, Naïve) and a previously developed method known as the Qualitative and Quantitative Multi-Model Predictor with kNN (QMMP + kNN) proposed by Lopez *et al.* (2015b)). This methodology is an improvement of the work of Quevedo *et al.* (2014).

Since the water demand time series is cyclic with a 24 hour period, we adopt the term of Naïve to refer the forecast performed as $Y_{t+1} = Y_{t-24}$. To clearly observe the accuracy of the BM compared with the Naïve method, Figure 8 shows a comparison between them.

The following tables (Tables 4–7) show the comparison between the proposed method, the QMMP + kNN, Holt–Winters, ANN, and Naïve method, for the flow meters p10012, p10017, p10025, p10031 of the Barcelona network.
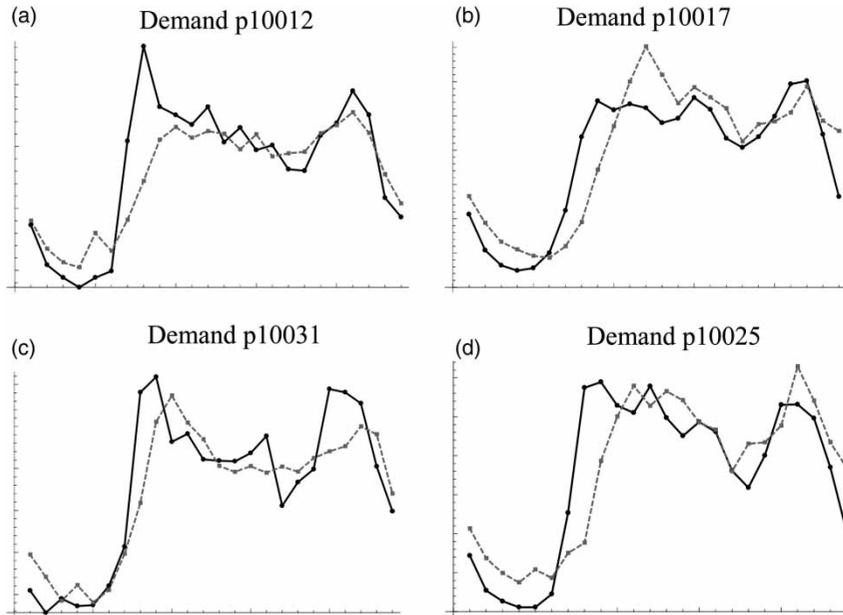
**Figure 6** │ Twenty-four hour steps' predictions of the demands p10012 (a), p10017 (b), p10025 (c), and p10031 (d). The solid line corresponds to the real data and the dashed line to the forecast.
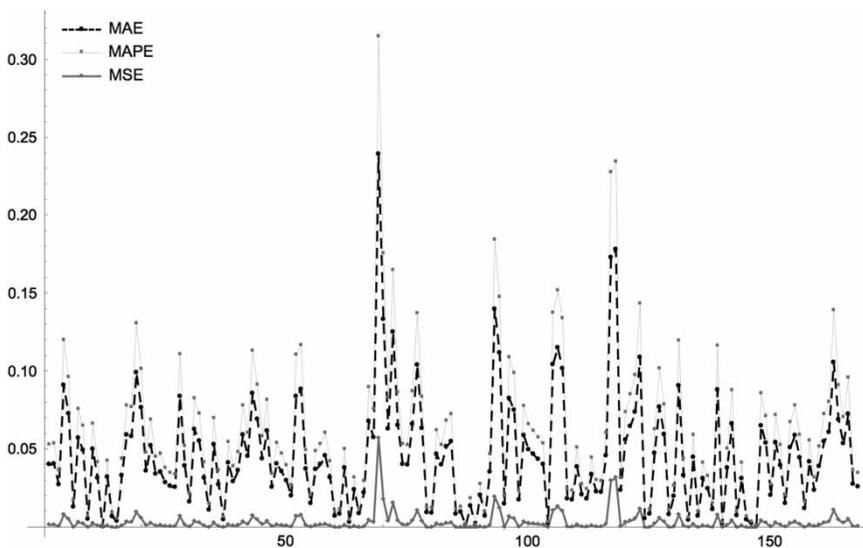


**Figure 7** │ Hour by hour comparison in term of MSE, MAE, MAPE error of the flow meter p10017.

From Tables 4–7, we can compare the results obtained with the BM and standard methods (ANN, Holt–Winters, Naïve). We can observe that the proposed method (BM) is more accurate for the considered application. The methods used for comparison re-inject the forecast data, to continue the forecast, in predictions of two steps and forward. Carrying on with this approach leads to increasing the

uncertainty. Thus, the longer is the forecast horizon, the bigger the forecasting error will be.

To observe this behavior, a detailed comparison of the forecasting results obtained with the re-injected ANN versus the BM is performed. In Figure 9, there are two columns of plots. The first column of Figure 9 corresponds to the forecasts obtained using the set of models and the
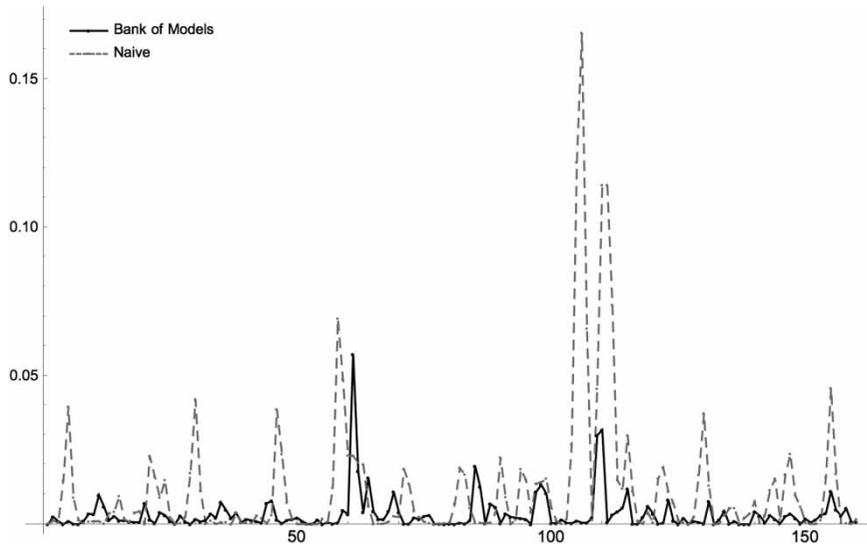
**Figure 8** │ Hour by hour comparison between the BM and the Naive method in terms of MSE error of the flow meter p10017.

**Table 4** │ Performance assessment in terms of MSE, MAE, MAPE, RMSE, with the methods QMMP, Holt–Winters, ANN, and Naïve of flow meter p10012

| Method | MSE | MAE | MAPE | RMSE |
|---|---|---|---|---|
| BM | 0.0098 | 0.0727 | 25.30% | 0.0991 |
| ANN | 0.0108 | 0.0767 | 22.21% | 0.1042 |
| Holt–Winters | 0.0156 | 0.0757 | 19.20% | 0.1078 |
| QMMP + kNN | 0.0073 | 0.0603 | 15.30% | 0.0790 |
| Naïve | 0.0137 | 0.0844 | 36.4% | 0.1173 |

**Table 6** │ Performance assessment in terms of MSE, MAE, MAPE, RMSE, with the methods QMMP, Holt–Winters, ANN, and Naïve of flow meter p10031

| Method | MSE | MAE | MAPE | RMSE |
|---|---|---|---|---|
| BM | 0.0076 | 0.0661 | 14.17% | 0.0872247 |
| ANN | 0.0131 | 0.0845 | 19.32% | 0.1144 |
| Holt–Winters | 0.0083 | 0.0660 | 14.16% | 0.0882 |
| QMMP + kNN | 0.0136 | 0.0863 | 18.51% | 0.1159 |
| Naïve | 0.0128 | 0.0810 | 17.37% | 0.11349 |

**Table 5** │ Performance assessment in terms of MSE, MAE, MAPE, RMSE, with the methods QMMP, Holt–Winters, ANN, and Naïve of flow meter p10025

| Method | MSE | MAE | MAPE | RMSE |
|---|---|---|---|---|
| BM | 0.0069 | 0.0656 | 17.57% | 0.0034 |
| ANN | 0.1198 | 0.2302 | 30.44% | 0.3461 |
| Holt–Winters | 1.7455 | 0.0992 | 26.57% | 0.1631 |
| QMMP + kNN | 0.0081 | 0.0738 | 19.77% | 0.0869 |
| Naïve | 0.0117 | 0.0773 | 23.09% | 0.108333 |

**Table 7** │ Performance assessment in terms of MSE, MAE, MAPE, RMSE, with the methods QMMP, Holt–Winters, ANN, and Naïve of flow meter 10017

| Method | MSE | MAE | MAPE | RMSE |
|---|---|---|---|---|
| BM | 0.0033 | 0.0452 | 11.90% | 0.0579 |
| ANN | 0.0063 | 0.0762 | 15.71% | 0.0793 |
| Holt–Winters | 0.0016 | 0.0253 | 6.64% | 0.0340 |
| QMMP + kNN | 0.0015 | 0.0238 | 5.83% | 0.0310 |
| Naïve | 0.0087 | 0.0697 | 18.35% | 0.0935 |

second one when the re-injected net is used. In the first column of Figure 9, the results obtained for the models corresponding to different prediction steps are shown: 8 (a), 16 (c), and 24 (e). In the second column, the plots correspond to results obtained with re-injected net for the same steps: 8 (b), 16 (d), and 24 (f) steps ahead. From this figure, it

can be noticed that when the forecasting horizon increases, the performance of the re-injected model becomes poorer compared to the one obtained with the BM.

However, if we compare the proposed method with the QMMP + kNN, we find similar accuracy among the four flow meters. The BM is more accurate in two of four flow
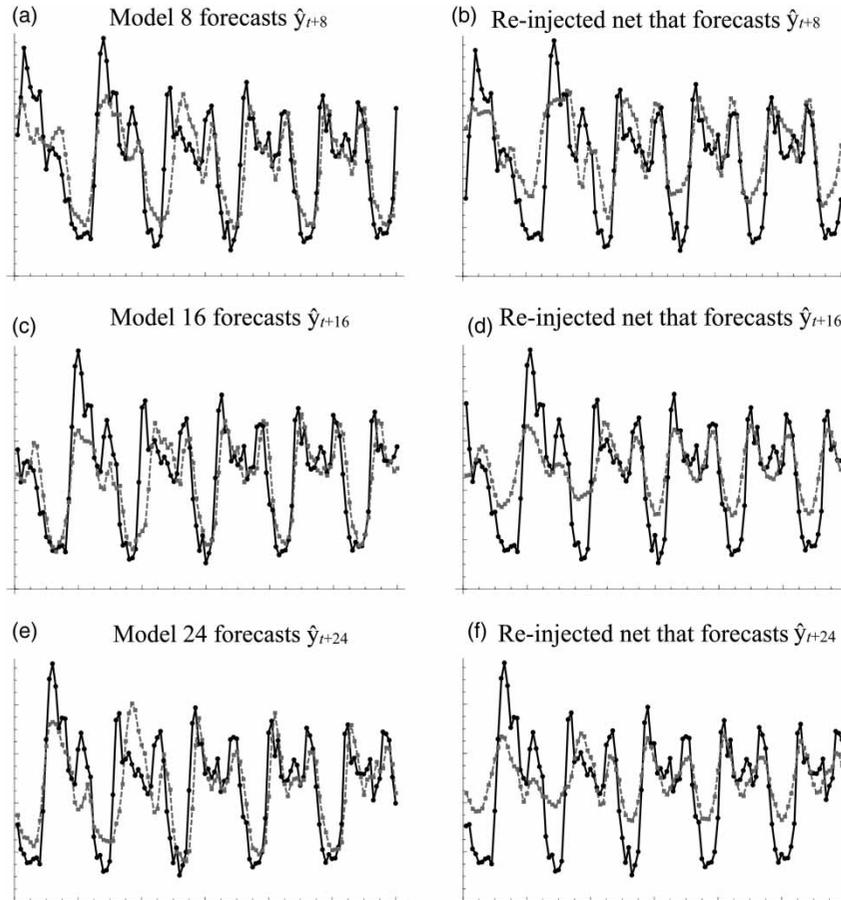
**Figure 9** │ The first column corresponds to the results obtained by the BM: Model 8 (a), Model 16 (c), and Model 24 (e). The second column shows the results obtained with the re-injected net: $\hat{y}_{t+8}$ (b), $\hat{y}_{t+16}$ (d), and $\hat{y}_{t+24}$ (f). If we compare the first column with the second one, we observe that the accuracy from the first column is better than the second column.

meters, and the accuracy difference is similar. Both methods use the modes to forecast, but use them in a different way; while the BM uses the next mode as one input, the QMMP + kNN uses the mode to define which prototype to select.

Finally, regarding the application of these forecasts to the operative management of water networks using MPC, it is very important to determine how uncertain the predictions are (worst-case scenario). A bad estimation can lead to higher production costs or the non-satisfaction of the demand supply. For this the reason, the confidence intervals (CI) of the forecasts provided by the BM are determined. This allows to bound the forecasting uncertainty.

The use of a BM facilitates the creation of the uncertainty bounds since the use of iterative prediction models will imply that the CI will grow exponentially with the number of steps forward to forecast. Figure 10 shows the

CI of the forecasted values for several flow meters using the proposed approach. The CI quantify the model uncertainty assuming that error follows normal distribution as follows:

$$\bar{x} - 1.96\frac{\sigma}{\sqrt{n}} < \mu < \bar{x} + 1.96\frac{\sigma}{\sqrt{n}} \tag{7}$$

where $\mu$ is interval with a 95% confidence, $\bar{x}$ is the mean of the error, $\sigma$ is the standard deviation, and $n$ is length of the population.

In Figure 10, each plot has four lines, the black line corresponds to the real data, the dotted line corresponds to the prediction, and the two dotted thinner lines (above and under the real data) defines the boundary of the confidence interval. With the CI, the MPC controller can know about
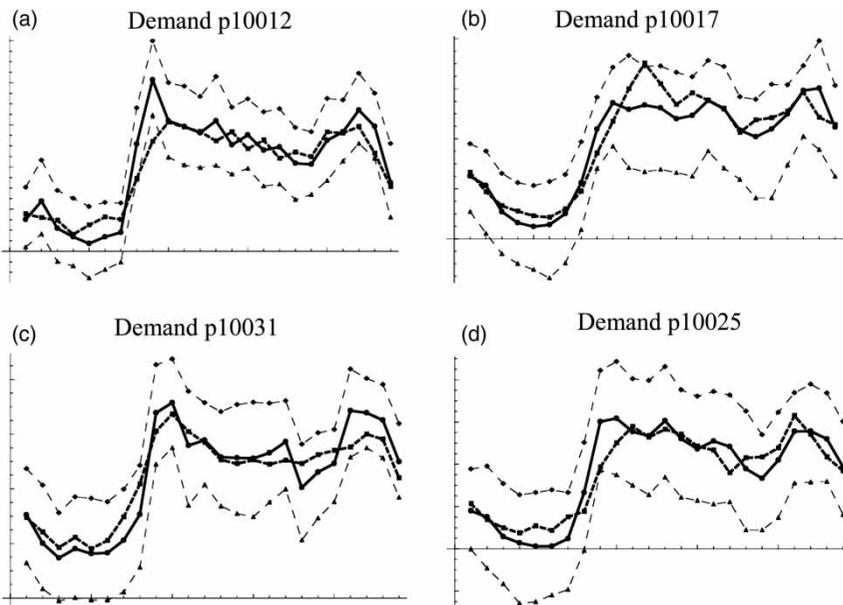
**Figure 10** | A qualitative perspective of our proposed approach. It shows 24 steps predictions of the demands p10012 (a), p10017 (b), p10025 (c), and p10031 (d). The continuous line corresponds to the real data and the dashed line to the forecast. The two dotted thinner lines define the boundary of the confidence interval.

the worst-case demand scenario, and prevent problems of demand satisfaction.

## CONCLUSIONS

When using MPC control for managing water distribution networks, the water demand forecast is very important in order to determine the best control sequence to apply to pumping stations and valves. With an accurate control sequence it is possible to reduce costs and improve economic and environmental benefits. In this work, we deal with the forecasting problem using a BM; where every model defined is independent and focused only in 1 hour in particular. We use ANN to define each model, and GA to obtain the best architecture, whereas traditionally this process is performed by trial and error. In this work, we propose the use of GAs to design the architecture of the net. For the training process, we start testing the experiments with traditional methods such as BP or BFGS, among others, and alternatively we use GA to define the connection weights, given the last method has a better accuracy than the traditional optimization methods. If we compare the time that the learning process using the traditional methods takes compared to GA, we found a huge difference

(1=minute to 45 minutes); but this process is performed off-line. Once the connection weights are obtained, the prediction is very fast.

In previous works, the short-term water demand forecast is performed considering the previous measurements, giving the opportunity to search if other factors like the temperature, humidity, human behavior, among others, have an influence on the water demand. In this work, we consider the human behavior factor by adding a mode to the input. The mode (or regime) concept comes from studying the time series, especially the ones that come from nature and the ones that came from human activities. Nature and humans have behavior patterns (like the weather in summer and in winter). With this assumption, we focus on detecting the pattern and predict it. Also, if an ANN has more information about the system, the more accurate the forecast is going to be. In our proposal, we identify the modes of the time series, we forecast the next mode and use it as an input of the ANN.

Finally, all the predictions are not completely accurate, there is always an error. The MPC controller needs to prevent the worst-case scenario, anticipating in advance the reaction in order to avoid problems of dissatisfaction of the water demand. The BM facilitates the construction of the confidence interval because in the forecast of the next

24 hours, we never consider estimated data to perform any forecast. In this work, the CI will allow determination of the worst-case scenario, so the MPC controller can anticipate in advance possible demand supply problems.

The system has been fully implemented in Python (Python Software Foundation 2016), using distributed EA (Fortin *et al.* 2012) (a Python plugin to perform GA) and the graphics showed in this article were performed in Mathematica (Wolfram Research 2015). The methodology was satisfactorily tested on the Barcelona water network.

As a future work, we plan to integrate this approach with the MPC controller and provide a robust MPC design methodology able to take into account the CI when computing the control actions to apply to the actuators. This idea can be complemented with the work of Giustolisi *et al.* (2014) that considers a Bayes-based method to estimate the predictive uncertainty based on the observed data. Also, it is interesting to include a statistical validation that allows detecting and minimizing structural errors following the analysis included in the work of Hutton & Kapelan (2015).

We could also consider performing a spatial analysis by consider other factors affecting the water demand, such as temperature, humidity, etc.

Finally, we could consider the use of the BM in the validation and reconstruction of data. By splitting the forecast into individual models, it will be easy to make a statistical analysis of the data that will allow to data validation and reconstruction to be carried out.

## ACKNOWLEDGEMENTS

## REFERENCES

Al-Hafid, M. S. & Al-Maamary, G. H. 2012 Short term electrical load forecasting using Holt-Winters method. *Al-Rafadain Engineering Journal* **20** (6).

Alperovits, E. & Shamir, U. 1977 Design of optimal water distribution systems. *Water Resources Research* **13** (6), 885–900.

Alvisi, S., Franchini, M. & Marinelli, A. 2007 A short-term, pattern-based model for water-demand forecasting. *Journal of Hydroinformatics* **9** (1), 39–50.

Bakirtzis, A., Petridis, V., Kiartzis, S., Alexiadis, M. & Maissis, A. 1996 A neural network short term load forecasting model for the Greek power system. *Power Systems, IEEE Transactions* **11** (2), 858–863.

Bakker, M., Vreeburg, J., van Schagen, K. & Rietveld, L. 2013 A fully adaptive forecasting model for short-term drinking water demand. *Environmental Modelling & Software* **48**, 141–151.

Benmouiza, K. & Cheknane, A. 2013 Forecasting hourly global solar radiation using hybrid k-means and nonlinear autoregressive neural network models. *Energy Conversion and Management* **75**, 561–569.

Chang, F.-J., Chiang, Y.-M. & Chang, L.-C. 2007 Multi-step-ahead neural networks for flood forecasting. *Hydrological Sciences Journal* **52** (1), 114–130.

Contreras, J., Espinola, R., Nogales, F. J. & Conejo, A. J. 2003 ARIMA models to predict next-day electricity prices. *IEEE Transactions on Power Systems* **18** (3), 1014–1020.

Cutore, P., Campisano, A., Kapelan, Z., Modica, C. & Savic, D. 2008 Probabilistic prediction of urban water consumption using the SCEM-UA algorithm. *Urban Water Journal* **5** (2), 125–132.

Donkor, E. A., Mazzuchi, T. A., Soyer, R. & Alan Roberson, J. 2012 Urban water demand forecasting: review of methods and models. *Journal of Water Resources Planning and Management* **140** (2), 146–159.

Flores, J. J., Loaeza, R., Rodríguez, H. & Cadenas, E. 2009 Wind speed forecasting using a hybrid neural-evolutive approach. In: *MICAI 2009: Advances in Artificial Intelligence*. Springer, Berlin, Heidelberg, Germany, pp. 600–609.

Flores, J. J., Rodriguez, H. & Graff, M. 2010 Reducing the search space in evolutive design of ARIMA and ANN models for time series prediction. In: *Advances in Soft Computing*. Springer, Berlin, Heidelberg, Germany, pp. 325–336.

Flores, J. J., Graff, M. & Rodriguez, H. 2012 Evolutive design of ARIMA and ANN models for time series forecasting. *Renewable Energy* **44**, 225–230.

Fortin, F.-A., De Rainville, F.-M., Gardner, M.-A., Parizeau, M. & Gagńe, C. 2012 DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research* **13**, 2171–2175.

Giustolisi, O., Brunone, B., Laucelli, D., Berardi, L., Campisano, A., Alvisi, S. & Franchini, M. 2014 16th water Distribution System Analysis Conference, WDSA2014 Assessment of the predictive uncertainty within the framework of water demand forecasting by using the model conditional processor. *Procedia Engineering* **89**, 893–900.

Hamilton, J. D. 1994 *Time Series Analysis,* vol. 2. Princeton University Press, Princeton, NJ, USA.

Harvey, A. C. 1990 *Forecasting, Structural Time Series Models and the Kalman Filter.* Cambridge University Press, Cambridge, UK.

Hippert, H. S., Pedreira, C. E. & Souza, R. C. 2001 Neural networks for short-term load forecasting: a review and evaluation. *IEEE Transactions on Power Systems* **16** (1), 44–55.

Hutton, C. J. & Kapelan, Z. 2015 A probabilistic methodology for quantifying, diagnosing and reducing model structural and predictive errors in short term water demand forecasting. *Environmental Modelling and Software* **66**, 87–97.

Jain, A., Kumar Varshney, A. & Chandra Joshi, U. 2001 Short-term water demand forecast modelling at IIT Kanpur using artificial neural networks. *Water Resources Management* **15** (5), 299–321.

Kumar, M. & Patel, N. R. 2010 Using clustering to improve sales forecasts in retail merchandising. *Annals of Operations Research* **174** (1), 33–46.

Liao, T. W. 2005 Clustering of time series data, a survey. *Pattern Recognition* **38** (11), 1857–1874.

Liao, T. W. 2007 A clustering procedure for exploratory mining of vector time series. *Pattern Recognition* **40** (9), 2550–2562.

Liu, J., Savenije, H. H. & Xu, J. 2003 Forecast of water demand in Weinan City in China using WDF-ANN model. *Physics and Chemistry of the Earth, Parts A/B/C* **28**, 219–224.

Lopez, R., Flores, J. J. & Puig, V. 2015a Multi-model forecasting based on a qualitative and quantitative decomposition with nonlinear noise filter applied to the water demand. In: *ROPEC 2015 (2015 IEEE International Autumn Meeting on Power, Electronics and Computing)*, Ixtapa, Mexico.

Lopez, R., Puig, V. & Rodriguez, H. 2015b An implementation of a multi- model predictor based on the qualitative and quantitative decomposition of the time-series. In: *ITISE 2015, International Work-Conference on Time Series*, Grenada, Spain, pp. 912–923.

Ma, Y., Borrelli, F., Hencey, B., Coffey, B., Bengea, S. & Haves, P. 2012 Model predictive control for the operation of building cooling systems. *IEEE Transactions on Control Systems Technology* **20** (3), 796–803.

Martinez Alvarez, F., Troncoso, A., Riquelme, J. & Riquelme, J. 2007 Partitioning-clustering techniques applied to the electricity price time series. In: *Intelligent Data Engineering and Automated Learning-IDEAL 2007*, Birmingham, UK, pp. 990–999.

McElroy, T. & Wildi, M. 2013 Multi-step-ahead estimation of time series models. *International Journal of Forecasting* **29** (3), 378–394.

Melgoza, J. J. R., Flores, J. J., Sotomane, C. & Calderón, F. 2004 Extracting temporal patterns from time series data bases for prediction of electrical demand. In: *MICAI 2004: Advances in Artificial Intelligence*, Mexico City, Mexico, pp. 21–29.

Nocedal, J. & Wright, S. J. 1999 *Numerical Optimization (Springer Series in Operation Research and Financial Engineering)*, vol. 43. Springer Science + Business Media.

Olszewski, R. T. 2001 *Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data*. Technical report, DTIC Document.

Ormsbee, L. E. & Lansey, K. E. 1994 Optimal control of water supply pumping systems. *Journal of Water Resources Planning and Management* **120** (2), 237–252.

Python Software Foundation 2016 Python language reference, version 2.7. Available at: http://www.python.org.

Quevedo, J., Puig, V., Cembrano, G., Blanch, J., Aguilar, J., Saporta, D., Benito, G., Hedo, M. & Molina, A. 2010 Validation and reconstruction of flow meter data in the Barcelona water distribution network. *Control Engineering Practice* **18** (6), 640–651.

Quevedo, J., Saludes, J., Puig, V. & Blanch, J. 2014 Short-term demand forecasting for real-time operational control of the Barcelona water transport network. In: *22nd Mediterranean Conference on Control and Automation*, University of Palermo, Palermo, Italy, pp. 990–995.

Romano, M. & Kapelan, Z. 2014 Adaptive water demand forecasting for near real-time management of smart water distribution systems. *Environmental Modelling & Software* **60**, 265–276.

Seron, M. M. & Braslavsky, J. H. 2000 Sistemas no lineales [Nonlinear systems]. Departamento de Electronica, Universidad Nacional de Rosario, Primer Cuatrimestre, Argentina.

Sociedad General de Aguas de Barcelona S.A. 2008 *Annual Corporate Governance Report*. Barcelona, Spain.

Sorjamaa, A., Hao, J., Reyhani, N., Ji, Y. & Lendasse, A. 2007 Methodology for long-term prediction of time series. *Neurocomputing* **70** (16), 2861–2869.

Stephenson, D. 1998 *Water Supply Management*. Kluwer Academic Publishers, Dordrecht.

Taieb, S. B., Sorjamaa, A. & Bontempi, G. 2010 Multiple-output modeling for multi-step-ahead time series forecasting. *Neurocomputing* **73** (10), 1950–1957.

Tiwari, M. K. & Adamowski, J. F. 2015 An ensemble wavelet bootstrap machine learning approach to water demand forecasting: a case study in the city of Calgary, Canada. *Urban Water Journal* 1–17.

van Overloop, P.-J. 2006 *Model Predictive Control on Open Water Systems*. IOS Press, Delft University Press, Amsterdam, The Netherlands.

William, W. & Shyong, W. 1994 *Time Series Analysis*. Addison-Wesley, Boston, MA, USA.

Wolfram Research 2015 *Mathematica 10.3*. Wolfram Research, Inc., Champaign, IL, USA.

World Health Organization 1993 *Guidelines for Drinking-Water Quality: Recommendations,* vol. 1. World Health Organization, Geneva, Switzerland.

Zhang, G., Patuwo, B. E. & Hu, M. Y. 1998 Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting* **14** (1), 35–62.

Zhou, S., McMahon, T., Walton, A. & Lewis, J. 2000 Forecasting daily urban water demand: a case study of Melbourne. *Journal of Hydrology* **236** (3–4), 153–164.