Leakage detection in water distribution networks using hybrid feedforward artificial neural networks

Hamideh Fallahi, Mohammadezra Jalili Ghazizadeh, Babak Aminnejad and Jafar Yazdi

ABSTRACT

Water leakage control in water distribution networks (WDNs) is one of the main challenges of water utilities. The present study proposes a new method to locate a leakage in WDNs using feedforward artificial neural networks (ANNs). For this purpose, two ANNs training cases are considered. For case 1, the ANNs are trained by average daily water demand, including small to large hypothetical leakages. In case 2, the ANNs are trained by hourly water demand and variable hourly nodal leakages over 24 hours. The training parameters are determined by EPANET2.0 hydraulic simulation software using MATLAB programming language. In both cases, first, ANNs are trained using flow rates of total pipes number. Then, sensitivity analysis is performed by hybrid ANNs for the flow rates of pipes number less than the number of the total pipes. The results of proposed hybrid ANNs indicate that if at least the flow rates of 10% of the total pipes are known (using flowmeters), then the leakage locations in both cases can be determined. Despite the complexity of case 2, because of the variations of demand and leakage over the 24-hour period, the proposed method could detect the leakage location with high accuracy.

Key words | feedforward artificial neural network, hourly water demand, leakage, variable hourly nodal leakage, water distribution networks

HIGHLIGHTS

- A leakage detection algorithm for WDNs using feedforward ANNs.
- ANNs were applied for two cases: average daily water demand and hypothetical leakages; variable water demand and nodal leakages over the 24-hour.
- ANNs training based on pipe flow rates and nodal leakage.
- A sensitivity analysis using Hybrid ANNs.
- The proposed algorithm successfully detects the leakage locations in both cases by the flow rates of 10% pipes.

This is an Open Access article distributed under the terms of the Creative Commons Attribution Licence (CC BY 4.0), which permits copying, adaptation and redistribution, provided the original work is properly cited (http://creativecommons.org/licenses/by/4.0/).

doi: 10.2166/aqua.2021.140
GRAPHICAL ABSTRACT

INTRODUCTION

Water resource is one of the most important necessities for human survival and development. Water distribution networks (WDNs), as critical infrastructure, play a vital role in safely delivering drinking water to customers. However, many networks have large water losses, burst pipes, and leakage due to aging, inadequate construction quality, or lack of maintenance, which further intensifies the water scarcity (Fazelabdolabadi & Golestan 2020; Zhao et al. 2020). The network’s overall efficiency is severely affected by the lack of proper maintenance. One of the prime reasons is leakage and illegal connections (Shahzad et al. 2013). WDNs suffer from leakages causing economic as well as social costs. There is a need for a platform to manage WDNs more efficiently by detecting, localizing, and controlling the leakages even before or as soon as they occur (Gupta & Kulat 2018).

In the last two decades, researchers performed a comprehensive analysis of efficient leakage management techniques to reduce water losses in WDNs. These techniques present different efficiency and precision of leakage detection in WDNs due to their properties. Pressure management in urban WDNs is one of the options that can significantly reduce water loss. Therefore, some researchers focus on the leakage reduction in WDNs based on pressure management techniques (Asgari & Maghrebi 2016; Zhao et al. 2020). To analyze the correlation between pressure and leakages, Greyvenstein & Van Zyl (2007) reported an experimental study to determine the leakage exponents for failed water pipes taken from the field and pipes with artificially induced leaks. Cassa & Van Zyl (2013) used finite element analysis to investigate the relationship between pressure head and leak area in pipes with longitudinal, spiral, and circumferential cracks. Ferraiuolo et al. (2020) provided an experimental and numerical computational fluid dynamics (CFD) investigation to assess the leakage-pressure relation for transversal rectangular orifices. On the other pressure management technique, Monsef et al. (2018) proposed an optimization code to estimate the instantaneous water demand based on the reported network pressures. De Marchis & Milici (2019) defined the relationship between the leak outflow, the total head at the leak, and other relevant parameters such as pipe stiffness, dimension, and shape of the leak. Acoustic sensors (working 24 hours per day) are the other object to locate the exact position of the leakages in different modes, fixed installation, and mobile exploration (Xue et al. 2020). Sarkamaryan et al. (2018) introduced an inverse transient analysis (ITA) modified Gaussian function to allocate candidate leakages in WDNs and also handle the simulation and measurement uncertainties.

Numerical approaches are important for modeling the leakages in WDNs. Tsanov et al. (2020) developed a method based on two Water Evaluation and Planning systems (WEAP) and MATLAB software packages. Mohapatra et al. (2014) investigated WDNs using EPANET for both intermittent and continuous water supply. Recent developments in computational intelligence, such as soft computing, machine learning, or data-driven modeling, help solve various problems in water resources domains.
These techniques are ideal for analyzing engineering problems where solutions need to develop without solving the microscale interactions that actually occur and may be poorly understood, but where measured data are readily available (Mounce et al. 2010; Yan et al. 2018). Artificial neural networks (ANNs) have been widely applied in different engineering fields and are modeling approaches based on how biological neural systems work. ANNs are models that learn the fundamental association between dependent and independent variables and estimate dependent variables using statistical learning algorithms (Jang et al. 2018). Artificial intelligence, mainly the multi-layer perceptron (MLP) ANNs, is applied widely to hydraulic engineering to estimate some parameters (Lima et al. 2018).

ANN-based models are effectively applied to leakage prediction. Stephen et al. (2007), using ANNs, introduced a method to analyze data from sensors measuring hydraulic parameters (flow and pressure) of the WDNs. Two neural architectures (static and time delay) were applied for time series pattern classification from detecting leakage. In the proposed method of Wachla et al. (2015), the location of leakage was determined using the group of neuro-fuzzy classifiers. The number of classifiers corresponds to the number of areas in which the network is divided. Attari et al. (2015) demonstrated the position and amount of leakage in WDNs using the combination of pressure and flow metering by ANNs. In other studies, Kang et al. (2018) proposed a novel and accurate leakage detection procedure with an adaptive design that fuses a one-dimensional convolutional neural network and a support vector machine (CNN-SVM). They also proposed a graph-based localization algorithm to detect the leakage location. Quiñones-Grueiro et al. (2018) discussed the use of supervised classifiers for leak location in WDNs. In the pattern recognition framework, four classification tools are widely used: nearest neighbor, bayes classifier, artificial neural networks, and support vector machines. Ma et al. (2019) proposed a novel method based on ANN and graph theory to detect and localize pipeline networks.

The ANN-based mentioned methods cannot detect small leakages because they require extensive ANN training data. Therefore, this matter leads to increases in training time and the complexity of the training process. Also, insufficient and inaccurate data in the ANNs training stage led to incorrect results (Moasheri & Jalili Ghazizadeh 2018).

This study proposes a new methodology to locate an hourly leakage in WDNs over 24 hours using feedforward ANNs. The main novelty is that the feedforward ANNs are trained based on the hourly variations of leakage and water demand over 24 hours. These hourly variations intensify the training process’s complexity and increase the training time of the feedforward ANNs. In this study, the proposed ANNs have high efficiency and can detect small and large leakages. The proposed ANNs are trained by using the pipes’ flow. The detection of the leakage location with a minimum number of pipes is essential to the proposed method’s practicality in WDNs. Therefore, in this study, new hybrid ANNs are introduced using sensitivity analysis to find the pipe’s optimal number.

The proposed ANNs are trained in two cases. For case 1, the ANNs are trained by average daily water demand, including small to large hypothetical leakages. In case 2, the leakage variations are considered over the 24-hour period. The water demand variations cause pressure fluctuations that lead to leakage variations. Thus, it is required to consider the relationship between pressure and leakage. Therefore, in case 2, the ANNs are trained by variable hourly water demand and nodal leakage over 24 hours. Then, in both cases, a sensitivity analysis is performed by hybrid ANNs to find the optimal pipe number. These hybrid ANNs are trained by flow rates of different percentages of pipes number. Also, the training parameters are determined by EPANET2.0 hydraulic simulation software and MATLAB programming language. The rest of this study is organized as follows: in the following section, the methodology, including the ANN definition and development, is presented. In the next section, the structure of the selected network is introduced. Next, the results and discussion presents two cases of the ANNs training process to detect the leakage location. In this section, the relationship between pressure and leakage in WDNs is investigated. Finally, the conclusion section is presented.

**METHODOLOGY**

**Artificial neural network (ANN) definition**

ANNs are currently known as practical tools for estimating and predicting the response of non-linear and multi-
parameter systems. These networks are based on learning from observed data sets and are widely used in solving different supervised learning problems (VaeziNejad et al. 2019). In the present study, a feedforward ANN is employed to construct a relationship between the input and output variables efficiently. Figure 1 illustrates the general structure of a feedforward ANN including an input layer, a hidden layer, and an output layer. The number of input variables determines the number of neurons in the input layer. The hidden layer determines the neural network’s main structure and will have the most significant effect on the network’s training method and its overall performance. Therefore, the number of neurons in the hidden layer is determined by minimizing a predefined error function over the training data by trial and error.

As shown in Figure 1, each neuron in the input layer accepts input, and each neuron s in the hidden layer or output layer accepts a set of inputs \(x_r\) from the neurons of the previous layer and produces an output \(y_s\) as follows (Zhou et al. 2019):

\[
y_s = \varphi \left( \sum_{r=1}^{m} w_{rs} x_r - \theta_s \right)
\]

where \(m\) represents the number of inputs, \(w_{rs}\) shows the connection weight of the \(r\)th input, \(\theta_s\) denotes the neuron’s threshold, and \(\varphi\) represents the activation function typically selected as the hyperbolic tangent function or sigmoid function.

After the structure of an ANN is fixed, in order to minimize the deviation of model outputs from actual outputs over training samples, the parameters, including all connection weights \(w_{rs}\) and all neuron biases \(\theta_s\), are tuned. This deviation can be determined based on the root mean squared error (RMSE) criterion:

\[
\text{minRMSE} = \sqrt{\frac{1}{M} \sum_{r=1}^{M} |y_s - \hat{o}_r|^2}
\]

where \(M\) is the number of samples, \((x_r, y_r)\) is the input-output pair of the \(r\)th sample, and \(\hat{o}_r\) is the actual output of the ANN on the input \(x_r\). ANN parameter optimization is usually considered a high-dimensional search problem. Typically, the values of weights and biases are restricted in the range of \([0, 1]\) or \([-1, 1]\).

In ANNs, RMSE is used as the performance function to determine the weights and biases. The RMSE criterion sometimes exhibits an acceptable value with proper convergence but does not satisfy the problem’s requirements. Indeed, the RMSE demonstrates the average behavior of the network, while the trained network may contain high errors for special points. If a good performance was also expected from the trained ANN for the worst data (with larger error), the RMSE may not be an appropriate residual metric. Therefore, to solve this problem, the maximum relative leakage error (MRLE) metric is defined in this study besides RMSE, which will be discussed later.

**ANN development (training and validation)**

As mentioned, the present study employs an ANN to detect the leakage location. At first, the ANN should be trained using a set of input and pre-specified output data. Then, the training data must be provided. For this purpose, a hydraulic model is run frequently under a significant number of synthetic (pipes flow and/or nodal pressure) scenarios to provide ANN training data. A hydraulic extended-period solver EPANET2.0, via the EPANET2.0 programmer’s toolkit, can simulate the studied network iteratively and obtain the required hydraulic parameters. EPANET2.0 is the most widely used software since the year 2000 (Rossman 2000). At first, a hypothetical nodal leakage is applied to typical nodes to produce the output variable’s values in the ANN training dataset. Also, the

![Figure 1](http://iwaponline.com/aqua/article-pdf/70/5/637/914109/jws0700637.pdf)
pipes’ flow obtained by the hydraulic solver is considered as the inputs of the ANN training dataset. All the ANN routines were coded in the MATLAB programming language. A toolkit is used to link MATLAB and EPANET2.0 software to simulate the pipes’ flow and nodal pressures.

Fitnet is a function in MATLAB software that is used for function fitting of ANN in this study. Function fitting is the process of training an ANN on a set of inputs to produce an associated set of target outputs. After selecting network architecture with the desired hidden layers and setting the number of neurons in the hidden layer by trial and error, it trains using a training dataset. By training the ANN, the values of biases and weights are obtained, and it can be used to generate outputs related to inputs of test data. The training method and the number of neurons in the hidden layer are important factors to achieve good performance. Levenberg–Marquardt backpropagation (Trainlm) is used as the network training function that updates weight and bias values according to Levenberg–Marquardt optimization. Trainlm is often the fastest backpropagation algorithm in the ANN MATLAB toolbox and is highly recommended as a first-choice supervised algorithm, although it does require more memory than other algorithms. Tansig (Tan-sigmoid) and Purelin are widely used as neural transfer functions in the hidden and output layers, respectively. Transfer functions calculate a neuron’s output from its net input. The values of the Tansig function are limited to the range of [−1, 1]. In the present study, a new parameter is defined as the ‘ScaleFactor’ to scale input and output parameters of training in the range of [−1, 1] to be more compatible with the transfer function.

STRUCTURE OF THE SELECTED NETWORK

In this study, the hypothetical rectangular network proposed by Poulakis et al. (2003) is utilized (Figure 2). This network consists of 50 pipes, 30 junction nodes, 20 loops, and one reservoir. An elevated reservoir supplies the water by gravity. The length of the vertical and horizontal pipes is 2,000 and 1,000 m, respectively, and the reservoir elevation is 52 m. In Figure 2, the range of pipe diameters is within 300–600 mm. The water demands are assumed at each node of the network through the water delivery system.

The value of 0.26 mm is considered for the piping-roughness coefficients for all pipes. In this study, the average daily water demand is assumed to be 20 L/s at each node (Poulakis et al. 2003).

RESULTS AND DISCUSSION

Case 1: ANN for leakage detection by average daily water demand and nodal leakage

According to the previous sections, the location of the leakages in WDN is determined using the ANN. The
proposed ANN can predict the location of a single leakage in a hypothetical WDN taken from Poulakis et al. (2003). In this case, the hypothetical leakage with average daily water demand at each node is defined as the output. The pipes’ flow is defined as the input to train the ANN. Figure 3 illustrates the training output matrix named leakage matrix. EPANET2.0 software generates the hydraulic parameter (pipe flow/pressure) using leakage matrix, which is required for ANN training. The average daily water demand at each node is considered to be 20 L/s. The hypothetical leakage is assumed based on a percentage of the total water demand in the hypothetical network. The range of hypothetical leakage is assumed to be 0.01 up to 1% of the total water demand. Therefore, the hypothetical leakage is considered from 0.06 up to 6.0 L/s with the step size of 0.11 L/s (0.06, 0.17, 0.28, … and 6.0 L/s). These values are added to the average daily water demand (20 L/s) for each node. This process leads to creating a $(30 \times 1,650)$ leakage matrix.

The first step for creating leakage matrix is a hypothetical leakage of 0.06 L/s applied in each of the 30 nodes in the hypothetical network to make a $(30 \times 30)$ matrix. Then, 0.11 L/s is added to the previous leakage, and the first procedure is iterated. This process continues until the final leakage of 6.0 L/s is reached, and leakage matrix is produced. All the processes are illustrated in Figure 3. This figure demonstrates the value and location of the nodal leakage for the network.

The training input matrix is generated by EPANET2.0 software. This software produces the pipes flow named pipes flow matrix $(50 \times 1,650)$ using leakage matrix. The columns of pipes flow matrix introduced which node has hypothetical leakage of 0.06–6.0 L/s. After determining pipes flow matrix as input and leakage matrix as output, the ANN is trained. The steps of training the ANN are demonstrated in Figure 4. After training the ANN, the leakage location can be determined using pipes flow.

As mentioned above under ‘Artificial neural network (ANN) definition’, besides the RMSE index, the relative leakage error (RLE) is also defined as below to investigate the ANN’s behavior:

$$RLE_i(\%) = \frac{CLD_{ANN_i} - HLD_i}{HLD_i} \times 100$$

where $RLE$ is relative leakage error, $CLD_{ANN}$ is calculated leakage and demand by ANN, $HLD$ is hypothetical leakage and demand, and $i$ is number of junctions, $i = 1, 2, 3, \ldots, 30$.

Using Equation (3), the RLE is calculated for the total nodes as the RLE matrix $(30 \times 1,650)$. This matrix’s columns represent the RLE values for the 30 nodes due to a single leakage at a junction. For example, the first 30 columns of the RLE matrix represent the relative error of hypothetical leakage of 0.06 L/s. As shown in Figure 3, this leakage is located in one of the nodes 1–30 in each column. Investigating the results indicates that in each column, the RLE matrix’s maximum value represents the leakage location.
Figure 4 | Flow chart of the proposed algorithm for the case-study network.
According to these results, MRLE can be expressed as:

\[ MRLE_j(\%) = \max (RLE_i)_j \]  

(4)

where \( MRLE \) is maximum relative leakage error, \( j \) is column of RLE matrix, and \( j \): 1, 2, 3, ..., 1,650.

The number of neurons in the hidden layer is an essential parameter for the ANN training. The optimal number of neurons in the hidden layer can be found based on the maximum of the MRLE (MaxMRLE). For this purpose, the steps explained in Figure 4 are employed for different neurons. This result is illustrated in Figure 5. As shown in Figure 5, the MaxMRLE value declines considerably for the neurons larger than 30 in training data (1650 cases). To verify this claim, the ANN must be implemented on validation data.

The nodal leakages apply in the range of 0.005% up to 1.2% of total water demand. The increased value of 0.21 L/s is used to construct the next value (0.03 L/s up to 7.05 L/s). By this procedure, the number of 1,050 cases is produced for validation data. Therefore, out of all data (2,700 samples) considered in this study, 60% is assigned to training data, and 40% is allocated to validation data. As mentioned, Figure 5 demonstrates the results of determining the optimal number of neurons. According to the above description and the results illustrated in Figure 5, the optimal number of neurons in the hidden layer is obtained as 30. Since the values of MaxMRLE for neuron numbers above 30 are very small, a magnification is performed in Figure 5 for the sake of clarification.

The MRLE values for different leakages from 0.06 to 6.0 L/s by considering 30 neurons at the hidden layer are

---

**Figure 5** The number of neurons at the hidden layer for the average daily water demand and nodal leakage (Blue: the training data (60% of all data), Brown: the validation data (40% of all data), Green and Red: the training and validation data for the neurons larger than 30, respectively). Please refer to the online version of this paper to see this figure in colour: http://dx.doi.org/10.2166/aqua.2021.140.
illustrated in Figure 6. Here, the y-axis represents the MRLE, and each 30th value on the x-axis introduces the hypothetical leakage position in one of the 30 nodes of the network. For example, the first 30 values represent a hypothetical leakage of 0.06 L/s in nodes 1–30, and the second 30 (31–60) indicate a hypothetical leakage of 0.17 L/s in nodes 1–30 (refer to Figure 3). This trend continues until the leakage of 6.0 L/s is reached, as demonstrated in Table 1. For better clarity, the x-axis in Figure 6 is divided into five categories based on the range of leakages illustrated in Table 1.

As shown in Figure 6, MRLE increases with increasing the hypothetical leakage from 0.06 to 6.0 L/s. However, the MRLE remains below 0.04% in total cases, which is an acceptable order (Asgari & Maghrebi 2016). Therefore, the trained ANN with as many inputs as the flow rates of total pipes number predicts the leakage location in WDN with acceptable accuracy. When the trained ANN is prepared to employ, the water network’s pipes flow should be measured by a flow meter. Using the pipes’ flow in trained ANN, the node with the maximum RLE (MRLE) represents the leakage location. The RLE value also indicates the deviation of the leakage value calculated by the ANN from the initial hypothetical leakage. According to Figure 6, this deviation has an acceptable range. The trained ANN also adequately covers leakages from those as small as 0.06 L/s to large ones.

Sensitivity analysis for Case 1

The results of the previous section indicate that the ANN training is performed based on a single leakage in nodes and flow rates of total pipes number. Indeed, the ANN is trained by considering flow rates of 100% of pipes number (50 pipes). Detecting the leakage location with a minimum number of pipes is desired in WDNs. For this purpose, in this section, the sensitivity analysis performs by hybrid ANNs for the flow rates of pipes number less than the number of the total pipes. The results are provided for the

![Figure 6](http://iwaponline.com/aqua/article-pdf/70/5/637/914109/jws0700637.pdf)

**Figure 6** | The calculated MRLE versus the location of the hypothetical leakage.
flow rates of different percentages of pipes number, including 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 70, 80, and 90% of the total pipe. Table 2 represents the percentage and the associated number of pipes in the network.

Sensitivity analysis was performed based on the secondary ANNs and, finally, hybrid ANNs. The secondary ANNs were trained by the flow rates of different percentages of pipes number as the input and flow rates of the total pipes number as the output. In each case, the number of the pipes select randomly. Since it is too time-consuming to consider all cases (for example, choosing 45 pipes from 50 pipes), the number of pipes is selected randomly in a limited iteration. Indeed, for each case, a percentage of all possible cases can be considered.

The outputs of the secondary ANNs are the input of the trained ANN by the flow rates of total pipes number. This procedure constructs hybrid ANNs (the same as hybrid mathematical functions). The hybrid ANNs receive the flow rates of different percentages of pipes number and detect the leakage location. The hybrid ANNs are built for different percentages of pipes number generally. Figure 7 illustrates the flowchart of this procedure to construct the hybrid ANNs. For example, for the hybrid ANN 20% total pipes number, a secondary ANN is trained by flow rates of 10 pipes as the input and flow rates of 50 pipes as the output. The output of this secondary ANN is used as the input of trained ANN by the flow rates of all pipes. Finally, the leakage location is obtained as a result of hybrid ANN. This process iterates for different percentages of pipes number. The proposed secondary ANNs are feedforward ANNs. The optimal number of neurons in each secondary ANN’s hidden layer varies according to the percentage of the pipes’ number. For the sake of brevity, details are not listed in this section.

After generating the hybrid ANNs, it is required to investigate which hybrid ANN can correctly detect the leakage location. This investigation is illustrated in Figure 8. In other words, it should be noted how many pipes can detect the leakage location.

As shown in Figure 8, two values display the leakage location in hybrid ANNs. Value ‘1’ denotes the percentage of pipes number that hybrid ANNs can successfully recognize the leakage location correctly. Value ‘0’ represents the percentage of pipes number that the hybrid ANNs cannot correctly locate the leakage. Thus, to detect the leakage location correctly by the proposed hybrid ANNs, the flow rates of at least 10% of the total pipes (five pipes) are required to be known.

The results presented in this section are obtained for the average daily water demand of 20 L/s and average nodal leakage in the range of 0.06–6 L/s. However, fluctuation in leakage value actually depends on the variation of the water demand. Therefore, the leakage variations should also be taken into account because of the variation of water demand value over the 24-hour period. Indeed, the water demand variations cause pressure fluctuations that lead to leakage variations over the 24-hour period. Thus, it is required to consider the relationship between pressure and leakage. This issue will be investigated in the next section.

| Table 1 | The hypothetical leakage values of the x-axis in Figure 6 |
|---------|----------------|
| x Axis  | 0–30 31–60 61–90 – 1,621–1,650 |
| Hypothetical leakage (L/s) | 0.06 0.17 0.28 – 6.0 |
| x Axis  | 0–330 331–660 661–990 991–1,320 1,321–1,650 |
| Range of leakage (L/s) | 0.06–1.16 1.27–2.37 2.48–3.58 3.69–4.79 4.90–6.00 |

| Table 2 | The percentage and associated number of pipes for the case-study network |
|---------|----------------|
| Percentage of pipes | 2 4 6 8 10 20 30 40 50 60 70 80 90 100 |
| Number of pipes | 1 2 3 4 5 10 15 20 25 30 35 40 45 50 |
Relationship between the pressure and leakage in WDNs

The dependency between the leakage and pressure is a well-known issue in WDNs. Thus, the leakage evaluation of a WDN requires the pressure fluctuations data over the 24-hour period. Lambert (1997) presented a good pattern of daily pressure fluctuation that is expected to have large values at night, small values in the morning and evening, and average values before noon and afternoon. Also, Lambert (1997) demonstrates that whenever the average pressure of a zone in a WDN changes over 24 hours (due to the water demand changes), the leakage rate fluctuates proportionally. Therefore, larger leakages are expected at hours when the pressure is high. Based on the investigation of the pressure variation over 24 hours through tests in Japan and the UK, the leakage-pressure relationship is proposed as follows (Gupta & Kulat 2018):

\[
L_1 = L_0 \left( \frac{P_1}{P_0} \right)^N
\]  

(5)

where \( L_1 \) and \( L_0 \) are leakages at the pressures \( P_1 \) and \( P_0 \), respectively. \( P \) is the pressure head of the zone, the indexes 0 and 1 are different pressure conditions, while \( N \) represents the pressure exponent changes from 0.5 to 2.5. Thornton & Lambert (2005) studied the coefficient of \( N \) for different projects and concluded that \( N \) could vary from 0.5 to 1 for small leakages in PVC pipes.

The present study employs water demand coefficients based on the water demand variations over 24 hours (the maximum hourly coefficient) using Standard No. 117-3 (IRIVPSPS 2013). Based on these coefficients and the average daily water demand of 20 L/s, the hourly water demand can be calculated. The results are illustrated in Table 3.

The nodal pressures can be calculated using hourly water demand from Table 3 and EPANET2.0 software. Then, using the nodal pressures, the variable hourly nodal leakage can be obtained by Equation (5). The variable hourly nodal leakage is calculated by the following assumptions:

1. The first pressure investigation performs at night (3–4 am) when the nodal leakage is maximum.
2. The maximum nodal leakage is assumed to be 1% of total water demand \((L_{\text{max}} = 6 \text{ L/s})\). This leakage is added to hourly water demand. The maximum nodal pressure \((P_{\text{max}})\) is calculated using EPANET2.0 software. Using \(L_{\text{max}}\) and \(P_{\text{max}}\) as the maximum nodal leakage and pressure and Equation (5) with \(N = 1\), Equation (6) becomes:

\[
L_t = L_{\text{max}} \left( \frac{P_t}{P_{\text{max}}} \right)
\]

where \(t\) shows the time measured in hours over the day \((t = 1, 2, 3, \ldots, 24)\), \(L_t\) and \(P_t\) are the hourly nodal leakage and nodal pressure at time \(t\), respectively.

For estimating the variable hourly nodal leakage at time \(t\) \((L_t)\) over 24 hours, it is required to calculate the hourly nodal pressure at this time \((P_t)\). For this purpose, first, a hypothetical leakage \((L_t')\) is assumed and is added to hourly water demand as provided in Table 3. Then, using EPANET2.0 software, the hourly nodal pressure is obtained \((P_t)\). Using this pressure \((P_t)\) and Equation (6), variable hourly nodal leakage at the time \(t\) \((L_t)\) is then calculated. This process repeats until the \(L_t\) and \(L_t'\) become equal. This process repeats to obtain the variable hourly nodal leakage over 24 hours at all nodes at the specific time \(t\).

All this calculation procedure was coded in the MATLAB programming language. Figure 9 depicts the 24-hour variations of the variable hourly nodal leakage. Each of the 30 diagrams exhibits the variable hourly nodal leakage variations at node 1 to node 30 over 24 hours.

**Case 2: ANN for leakage detection by hourly water demand and variable nodal leakage over 24 hours**

The proposed ANN in Case 1 above is trained based on the average daily nodal leakage and water demand. According to Table 3 and Figure 9, the water demands and leakages at all nodes change hourly over 24 hours. Thus, in this case, the ANN should be trained by considering these variations over 24 hours. The first step of ANN training is to determine the training inputs and outputs. Table 3 (hourly water demand) and Figure 9 (variable hourly nodal leakage) are used to generate the leakage matrix as the training output. For this purpose, at the first hour, hourly water demand from Table 3 and variable hourly nodal leakage from Figure 9 are included in each node (1–30 nodes), and finally, a \((30 \times 30)\) matrix is generated. This process repeats for 24 hours. Then, 24 of the generated \((30 \times 30)\) matrices are put next to each other to create a \((30 \times 720)\) leakage matrix (like Figure 3). Leakage matrix is utilized as the output data for ANN training. Using leakage matrix and EPANET2.0 software, pipes flow matrix \((50 \times 720)\) is generated as input data for ANN training. After creating the input and output data of the ANN training, the optimal number of neurons in the hidden layer should be determined. This task is illustrated in Figure 10.

It is required to generate validation data to find the optimal number of neurons in the hidden layer. For this purpose, the following assumptions are made:

1. Hourly water demand follows a normal distribution over 24 hours.
2. Hourly water demand is considered as the mean normal distribution.
3. Variation coefficient is equal to be 10%.

The hourly water demand for validation data is generated using mentioned assumptions and Table 3 data. At each time, 10% of hourly water demand must be added to the hourly water demand value. Indeed, all the hourly

---

Table 3 | Hourly water demand variations for large cities (RIIVPSPS 2013)

| Time (hour) | Hourly water demand coefficient | Hourly water demand (L/s) | Time (hour) | Hourly water demand coefficient | Hourly water demand (L/s) |
|------------|---------------------------------|---------------------------|------------|---------------------------------|---------------------------|
| 0–1        | 0.5                             | 10                        | 12–13      | 1.45                            | 29                        |
| 1–2        | 0.45                            | 9                         | 13–14      | 1.2                             | 24                        |
| 2–3        | 0.35                            | 7                         | 14–15      | 1.15                            | 23                        |
| 3–4        | 0.3                             | 6                         | 15–16      | 1.05                            | 21                        |
| 4–5        | 0.45                            | 9                         | 16–17      | 1.1                             | 23                        |
| 5–6        | 0.75                            | 15                        | 17–18      | 1.4                             | 28                        |
| 6–7        | 1.1                             | 22                        | 18–19      | 1.65                            | 33                        |
| 7–8        | 1.25                            | 25                        | 19–20      | 1.8                             | 36                        |
| 8–9        | 1.1                             | 22                        | 20–21      | 1.45                            | 29                        |
| 9–10       | 1                               | 20                        | 21–22      | 1.1                             | 22                        |
| 10–11      | 0.9                             | 18                        | 22–23      | 0.8                             | 16                        |
| 11–12      | 1.1                             | 22                        | 23–24      | 0.6                             | 12                        |
water demand in Table 3 should be multiplied by coefficient 1.1. This process creates the hourly water demand of validation data. Using Equation (6) and the mentioned procedure in the previous section, the variable hourly nodal leakages in all nodes are produced for validation data (the maximum nodal leakage at each node is assumed 1.2% of total water demand for validation data). Then, the pipes flow for validation data are calculated using the variable hourly nodal leakage and EPANET2.0 software. Finally, inputs and outputs of validation data serve to trained ANN to determine the hidden layer’s optimal neurons. Figure 10 depicts these results. As shown in Figure 10, the MRLE (MaxMRLE) maximum based on different neurons number considerably reduces after 30 neurons. Therefore, 30 neurons can be claimed to be the optimal number for this case’s hidden layer. The hourly variations of the leakage and water demand increase the problem’s complexity. However, due to the high efficiency of the proposed ANN, 30 neurons are enough for this case.

According to Figure 10 and the above-mentioned explanation, an ANN with 30 neurons can be proposed to predict the leakage over 24 hours. MRLE value can also be calculated, which is demonstrated in Figure 11. Figure 11 shows that the MRLE value is below 0.1% over 24 hours. In this case, this value is slightly larger than Case 1 with average daily water demand and nodal leakage value (below 0.04%). As can be seen, the proposed ANN with as many inputs as the flow rates of total pipes number predicts the leakage location with acceptable accuracy (Asgari & Maghrebi 2016). Thus, to employ the trained ANN, pipes’ flow rates should be measured by some flow meters. Then, using the proposed ANN, RLE values are calculated at

![Figure 9](https://example.com/figure9.png)

**Figure 9** The variable hourly nodal leakage over 24 hours for all 30 nodes.
each node. Finally, the node with the maximum RLE represents the leakage location over 24 hours.

In Figure 11, the x and y axes represent the hours and MRLE for each node, respectively. The MRLE variations in each time step on the x-axis represent the MRLE value in the 30 nodes (1–30 nodes). For better clarity, two specific times are magnified. In each time step, 30 MRLE values are estimated and finally 720 values are generated on the diagram for 24 hours. As shown in Figure 11, the maximum MRLE occurs in the fourth time step (the interval between 3.00 and 4:00 on the x-axis). At this time, the network experiences the lowest water demand and highest leakage. Another critical time in WDNs is when the highest water demand and lowest leakage occur. Usually, in typical WDNs, this happens at 20:00 (the interval between 19:00 and 20:00 on the x-axis). These critical times (4:00 and 20:00) are illustrated in separate diagrams in Figure 11.

Sensitivity analysis for Case 2

Similar to Case 1, in this case, the ANN is trained with the flow rates of 100% of pipes number (50 pipes). However, a sensitivity analysis is required to reduce the number of the required pipes. The results are provided for the flow rates of different percentages of pipes number, including 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 70, 80, and 90% of the total pipes. Thus, hybrid ANNs (like Figure 7) should be created to detect the leakage location by flow rates of different pipe number percentages. In this case, the number of the pipes for secondary ANNs is selected randomly, similar to Case 1. The results of this case are similar to those of Case 1 (Figure 8). The proposed hybrid ANNs can predict the leakage location correctly by measuring the flow rates of at least 10% of pipes number (five pipes). The fewer pipes number could not detect the leakage location. Obviously, the
training inputs and outputs of the Hybrid ANNs in Case 2 are more complex than Case 1. However, training parameters in Case 2 are more compatible with real-world problems. The minimum number of pipes to detect the leakage location is the same for both cases. Therefore, although the leakage and water demand changes over 24 hours in Case 2 and the complexity of the problem increased, because of the high efficiency of the proposed ANN, it is still possible to detect the leakage location with the number of the minimum pipe.

CONCLUSIONS

The present study proposes a new methodology that considers variable leakage over 24 hours to find the leakage location in WDNs using feedforward artificial neural networks (ANNs). The proposed ANNs were studied for two cases with an optimal neuron number of 30 in the hidden layers. The relation between leakage and pressure was implemented over 24 hours. Based on these results, the variable hourly nodal leakage over 24 hours at all nodes was obtained. As a result, the accuracy of the trained ANNs by flow rates of total pipes in Case 1 and Case 2 was obtained below 0.04 and 0.1%, respectively. However, for the applicability of the proposed method in real WDNs, the sensitivity analysis was performed by hybrid ANNs for the fewer pipe numbers with known flow rates. The results showed that if at least flow rates of 10% of the total pipe numbers are known by flowmeters, the leakage location can be found accurately. Although the leakage and water demand were changed hourly over 24 hours in Case 2, showing the complexity of the problem, it could detect the leakage location because of the high efficiency of the proposed hybrid ANNs. The obtained results indicate the promising applicability of the proposed methodology for finding leakage location in WDNs. However, it should be noted that for WDNs with a very large number of pipes, the computational cost of the training procedure may become a drawback.
DATA AVAILABILITY STATEMENT

All relevant data are included in the paper or its Supplementary Information.

REFERENCES

Asgari, H. R. & Maghrebi, M. F. 2016 Application of nodal pressure measurements in leak detection. Flow Measurement and Instrumentation 50, 128–134.

Attari, M., Zaki Pour, M. & Maghrebi, M. F. 2015 Determination of the position and amount of leakage in water supply networks using the combination of pressure metering and flow metering by neural networks. In: 2nd National Conference of Ways Crisis in Iran and the Middle East, Shiraz, Iran (in Persian).

Cassa, A. M. & van Zyl, J. E. 2015 Predicting the head-leakage slope of cracks in pipes subject to elastic deformations. Journal of Water Supply: Research and Technology-AQUA 62, 214–223. doi:10.2166/aqua.2013.094.

De Marchis, M. & Milici, B. 2009 Leakage estimation in water distribution network: effect of the shape and size cracks. Water Resources Management 53, 1167–1183. doi:10.1007/s11269-018-1985-6.

Fazelabdalolabadi, B. & Golestan, M. H. 2020 Towards Bayesian quantification of permeability in micro-scale porous structures. The Database of Micro Networks, HighTech and Innovation Journal 1 (4). doi:10.28991/HIJ-2020-01-04-02.

Ferraiuolo, R., De Paola, F., Fiorillo, D., Caroppi, G. & Pugliese, F. 2020 Experimental and numerical assessment of water leakages in a PVC-A pipe. Water 12 (6), 1804. doi:10.3390/w12061804.

Greyvenstein, B. & van Zyl, J. E. 2007 An experimental investigation into the pressure-Leakage relationship of some failed water pipes. Journal of Water Supply: Research and Technology-AQUA 56, 117–124. doi:10.2166/aqua.2007.065.

Gupta, A. & Kulat, K. D. 2018 A selective literature review on leak management techniques for water distribution system. Water Resources Management 32, 3247–3269. doi:10.1007/s11269-018-1985-6.

Islamic Republic of Iran Vice Presidency for Strategic Planning and Supervision (IRIVPSS) 2015 Design Criteria of Urban and Rural Water Supply and Distribution Systems. No. 117-3. Available from: http://seso.moc.org.ir (in Persian).

Jang, D., Park, H. & Choi, G. 2018 Estimation of leakage ratio using principal component analysis and artificial neural network in water distribution systems. Sustainability 10, 750. doi:10.3390/su10030750.

Kang, J., Park, Y. J., Lee, J., Wang, S. H. & Eom, D. S. 2018 Novel leakage detection by ensemble CNN-SVM and graph-based localization in water distribution systems. IEEE Transactions on Industrial Electronics 65 (5), 4279–4289.

Lambert, A. 1997 Pressure management/leakage relationships: theory, concepts, and practical applications. In: Proceedings of Minimizing Leakage in Water Supply/Distribution Systems. IQPC Seminar, London.

Lima, G. M., Brentan, B. M., Manzi, D. & Luvizotto Jr., E. 2018 Metamodel for nodal pressure estimation at near real-time water distribution systems using artificial neural networks. Journal of Hydroinformatics 20 (2), 486–496.

Ma, D., Wang, J., Sun, Q. & Hu, X. 2019 A novel broad learning system-based leakage detection and universal localization method for pipeline networks. IEEE Access 7, 42343–42353. doi:10.1109/ACCESS.2019.2908015.

Moasheri, R. & Jalili Ghazizadeh, M. R. 2018 Identifying position and amount of two simultaneous leaks in water supply networks by a two-step algorithm. Amirkabir Journal of Civil Engineering 52 (5), 1–10. doi:10.22060/cej.2018.15272.5867 (In Persian).

Mohapatra, S., Sargaonkar, A. & Labhasetwar, P. K. 2014 Distribution network assessment using EPANET for intermittent and continuous water supply. Water Resources Management 28 (11), 3745–3759.

Monsef, H., Naghashzadegan, M., Farmani, R. & Jamali, A. 2018 Pressure management in water distribution systems in order to reduce energy consumption and background leakage. Journal of Water Supply: Research and Technology-AQUA 67 (4), 397–403. doi.org/10.2166/aqua.2018.002.

Mounce, S. R., Boxall, J. B. & Machell, J. 2010 Development and verification of an online artificial intelligence system for detection of bursts and other abnormal flows. Journal of Water Resources Planning and Management 136 (3), 309–318. doi:10.1061/(ASCE)WR.1943-5452.0000030.

Poulakis, Z., Valougeorgis, D. & Papadimitriou, C. 2005 Leakage detection in water pipe networks using a Bayesian probabilistic framework. Probabilistic Engineering Mechanics 18, 315–327.

Quíñones-Grueiro, M., Lázaro, J. M., Verde, C., Prieto-Moreno, A. & Orestes Llanes-Santiago, O. 2018 Comparison of classifiers for leak location in water distribution networks. IFAC PapersOnLine 51 (24), 407–413.

Rossman, L. A. 2000 Epanet 2 User Manual. Water Supply and Water Resources Division National Risk Management Research Laboratory, Cincinnati, OH, EPA/600/R-00/057.

Sarkamaryan, S., Haghhighi, A. & Adib, A. 2018 Leakage detection and calibration of pipes networks by the inverse transient analysis modified by Gaussian functions for leakage simulation. Journal of Water Supply: Research and Technology-AQUA 67 (4), 404–413. doi.org/10.2166/aqua.2018.176.

Shahzad, G., Rehan, R. & Fahim, M. 2019 Rapid performance evaluation of water supply services for strategic planning. Civil Engineering Journal 5 (5), 1197–1204. doi: 10.28991/cej-2019-03091324.

Stephen, R., Mounce, S. R. & Machell, J. 2007 Burst detection using hydraulic data from water distribution systems with artificial neural networks. Urban Water Journal 3 (1), 21–31. doi: 10.1080/15730620600578538.
Thornton, J. & Lambert, A. 2005 Progress in practical prediction of pressure: leakage, pressure: burst frequency, and consumption relationships. In: Proceedings of IWA Special Conference Leakage, Halifax, Canada, pp. 12–14.

Tsanov, E., Ribarova, I., Dimova, G., Ninov, P., Kossida, M. & Makropoulos, C. 2020 Water stress mitigation in the Vit river basin based on WEAP and matLab simulation. Civil Engineering Journal 6 (11), 2058–2071. doi:10.28991/cej-2020-03091602.

VaeziNejad, S. M., Marandi, S. M. & Salajegheh, E. 2019 A hybrid of artificial neural networks and particle swarm optimization algorithm for inverse modeling of leakage in earth dams. Civil Engineering Journal 5 (9), 2041–2057. doi:10.28991/cej-2019-03091392.

Wachla, D., Przystalka, P. & Moczulski, W. 2015 A method of leakage location in water distribution networks using artificial neuro-fuzzy system. IFAC-PapersOnLine 48 (21), 1216–1223. doi:10.1016/j.ifacol.2015.09.692.

Xue, Z., Tao, L., Fuchun, J., Riehle, E., Xiang, H., Bowen, N. & Singh, R. P. 2020 Application of acoustic intelligent leak detection in an urban water supply pipe network. Journal of Water Supply: Research and Technology-AQUA 69 (5), 512–520. doi:10.2166/aqua.2020.022.

Yan, Y., Wang, L., Wang, T., Wang, X., Hu, Y. & Duan, Q. 2018 Application of soft computing techniques to multiphase flow measurement: a review. Flow Measurement and Instrumentation 60 (1), 30–43. S0955-5986(17)30465-X. doi:10.1016/j.flowmeasinst.2018.02.017.

Zhao, M., Zhang, C., Liu, H., Fu, G. & Wang, Y. 2020 Optimal sensor placement for pipe burst detection in water distribution systems using cost-benefit analysis. Journal of Hydroinformatics 22 (3), 606–618.

Zhou, X. H., Zhang, M. X., Xu, Z. G., Cai, C. Y., Hung, Y. J. & Zhang, Y. J. 2019 Sallow and deep neural network training by water wave optimization. Swarm and Evolutionary Computation 50, 100561.

First received 14 October 2020; accepted in revised form 1 May 2021. Available online 21 May 2021