A hybrid model-based method for leak detection in large scale water distribution networks

No name manuscript No.

(will be inserted by the editor)

A Hybrid Model-Based Method for Leak Detection in Large Scale Water Distribution Networks

Zahra Fereidooni · Hooman Tahayori * · Ali Bahadori-Jahromi

Received: date / Accepted: 12 June 2020

Abstract During the past decades, the problem of finding leaks in Water Distribution Networks (WDN) has been controversy. The quicker detection of leaks prevents water loss and helps avoiding their economic and environmental consequences. On the other hand, increasing the speed of leak detection increases the false leak detection that imposes high costs. In this paper, we propose a real-time hybrid method using AI algorithms and hydraulic relations for detecting and locating leaks and identifying the volume of losses material. The proposed method relies on simple and cost-effective flow sensors installed on each junction in the pipeline network. We demonstrate how influential features for leak detection would be generated by using hydraulic equations like Hazen-Williams, Darcy-

Zahra Fereidooni
Department of Computer Science & Engineering and IT, Shiraz University, Shiraz, Iran
E-mail: zahra.fereydooni@shirazu.ac.ir

Hooman Tahayori
Department of Computer Science & Engineering and IT, Shiraz University, Shiraz, Iran
* corresponding author
Tel.: +98-71-3613 3544
E-mail: tahayori@shirazu.ac.ir

Ali Bahadori-Jahromi
School of Computing and Engineering, University of West London,
London UK, W5 5RF UK,
E-mail: Ali.Jahromi@uwl.ac.uk

Weisbach and pressure drop. Through exploiting Decision Tree, KNN, random forest, and Bayesian network we build predictive models and based on the pipeline topology, we locate leaks and their pressure. Comparing the results of applying the proposed method on various leak scenarios shows that the proposed method in this paper, outperforms other existing methods.

Keywords: WDN, Leak, Flow, Pressure, Machine Learning

1 Introduction

Pipeline systems, either single pipe or a pipe network, may transport oil, gas, water, etc. Leaks in pipes may occur due to improper pipe material, weak joints, earth movement, internal corrosion, corrosive soils, construction or utility digging, seasonal changes in temperature, heavy traffic load, tidal influence, water hammer, air entrapment, and so on. Leakage detection in pipeline networks is important because of the value of the material which flows in the pipe and the possible environmental damages that a leak may cause. Despite the importance of leakage detection and localization, still not all its related effects and aspects are studied [1]. In particular, leakage in WDNs may cause water scarcity, ground subsidence, and sinkholes that can be life threatening [2][3].

Many different methods for leakage detection are proposed that can be put in three basic categories. The first category consists of the methods that relies on gathering data from acoustic instruments [4], camera, ground penetrating radar (GPR), fiber optic [5] and alike. The main problem with such methods can be counted as scalability, installation and maintenance costs and high power consumption.

Van Hieu *et al.* in [6] used wireless sensors and external acoustic instruments to detect leaks. Their method mainly relies on the fact that when a leak happens in a pipe an acoustic sound occurs. However, noise may drastically affect their system. Moreover, its time complexity is high and is expensive to deploy. Khalif *et al.* in [7] used an internal acoustic instrument named "hydrophone". Hydrophones are pushed into the pipe network and move with the flow. They send the acoustic waves through the sensors that are deployed on the pipes. Applicability of this method depends on the pipe diameters and material. Moreover, applying this method over a large-scale water network is expensive and processing its generated signals is time consuming. Huang *et al*. [8] in 2007 presented a method that uses fiber optic as a tool for gathering information. In each pipe, an acoustic pressure induces an optical phase signal on the optical fiber fixed on the surface of the pipe. By finding phase differences from two points with the same length, leaks are detected. This system is expensive and when a part of a fiber optic encounters a problem, the whole pipe cannot be monitored. Sinha *et al.* [4] proposed three different methods through using vision based systems to detect leaks. Similar to closed circuit television (CCTV) their proposed method monitors inside a pipe. Another vision based method they proposed, uses laser that is deployed on top of the pipeline. The third method is in the form of vision based ultrasonic inspection – Sonar - in which a beam of very high frequency coherent sound energy, which is above the human hearing range, is used. The sound waves travels into the object that is to be inspected. Based on the reflection of the sound waves, issues with the pipe can be detected. Ultrasonic wave reflects most easily when it crosses an interface between two materials that are perpendicular to the wave. Hence, cracks that lie perpendicular to the wave are easily detected, but cracks that lie parallel to the beam usually are not identified by an ultrasonic examination. Evaluation of this method is often difficult. The main idea of Hunaidi *et al*. in [9] is using ground

penetrating radars for leak detection. However, different types of the soil may affect the signal penetration. Cody *et al.* in [10] have applied autoencoders on spectrograms of collected hydroacoustic data for leak detection in WDNs. The method however, is vulnerable to noise level of the baseline system. In [11]a pipe-in-pipe design integrated with wireless information and communication network is proposed for leak prevention and detection. Interpipe space in this method should be filled with insulation filler that has high wireless communication signal conductivity. In case of leakage from inner pipe, wireless communication within the interpipe space would be degraded and leak would be detected and located. Although this method can provide leakless pipeline, however it is too costly. In [12] vibration sensors are employed and using SVM leakage is detected. Although the method is efficient, however due to the limited effective range of the sensors, a large number of sensors should be used in a WDN. Noises (e.g. those that are generated by opening water-taps) that normally occur in an urban WDN, may negatively affect the performance of the proposed method.

Methods in the second category are transient-based which are more popular since transient flow may bring more information when a leak happens in a network. The main idea behind these methods is comparing transient hydraulic parameters that are collected from sensors with the calculated parameters of the steady-state equations. Observing the differences between collected and calculated parameters helps identifying abnormal situations.

Al-khomairi in [13] by trial and error found the minimal squares of deviations between observed and computed pressure by the equation of motion and continuity pipe. However, he considered only a single pipe and ignored noises. Taylor *et al.* in [14] presented a procedure that utilizes transient state pressure to detect leakage in piping systems. Transient flow, causes by opening or closing a valve, is analyzed in the time domain by the method of characteristics, which is widely used in solving the hyperbolic partial differential equations that describe transient-state flow in pipes. The results are then transformed into frequency domain by the fast Fourier transform. This method is used to develop a frequency response diagram at the valve end. It can be used for comparing the frequency response diagram of a modeled system without leaks with the frequency response diagram developed by gradually opening or closing a valve at the downstream. Although diagram checking instead of using hydraulics equation is a good idea but noises in the background may affect the diagrams which demands for more time to check and validate. In [15] it is shown that detecting leaks based on leak-reflected signals, while there is noise or leak is slight is not applicable. Instead, the use of leak-induced damping which is less sensitive to noise is proposed. However it is shown that the accuracy of this method is even lower than other damping based methods [15].

Other methods that can be put in the third category, rely on hydraulic sensor data and can be referred to as real-time leak detection methods. Shorter time intervals and low power consumption to collect and transfer data can be counted as the advantages of these methods. Sensors used in these methods are cheaper and more flexible to deploy than the sensors of the methods discussed in the previous categories. Data would be collected from different types of sensors like underground wireless sensor or smart meters. In real-time monitoring, quantitative parameters like flow and pressure and qualitative parameters such as turbid would be accessible through related sensors. Various sensors for measuring flow, temperature, pressure and turbid exist at reasonable prices. Depending on the application and the required data, sensors' costs, installation and maintenance fees, one or more of such sensors would be exploited. The main idea behind these methods is using

the historical data gathered by sensors to predict future values of parameters by data mining models.

Cuguer´o-Escofet *et al.* in [16] developed a model-based method for leak detection relying on Mont-Carlo. Soldevilla *et al.* in [17], [18] proposed a method for leak detection in Water Distribution Networks (WDNs) based on the use of classifiers (KNN, Bayesian) and pressure models. In their proposed method first, pressure data is gathered. Then, in the second step, a classifier (K-NN and Bayesian) is applied to obtain residuals with the aim of determining the leak locations. They claim that by using the recursive method, obtained results would be improved in shorter time since data can be kept up-to-date as the new data is available. The method was tested using the data generated by a simulator and the tests were conducted in different scenarios with various pressures and classifiers.

Mashford *et al.* in [19] described a method for detecting and sizing leaks in a pipe network by processing pressure values obtained from a number of sensing points in the network using Support Vector Machines (SVMs). Buchberger *et al.* in [20] presented a method for detecting leaks in the residential service zone of WDNs that uses continuous measurements of flow rate during the low water use time (winter nights). This method uses statistical analysis based on computing the mean and standard deviation of measured flow. There are limitations for implementing this algorithm like possibility or impossibility of recording data in short time intervals e.g. seconds. Moreover, this method can only detect if leak has happened in the WDN but cannot recognize the number and placement of the leaks. Mazzolani *et al.* in [21] proposed a method for estimating leakage in WDNs by using flow parameters. They presented a bottom-up methodology for leakage assessment in WDNs, based on a physically consistent formulation of the nonlinear regression problem and using only WDN inflow readings in a data assimilation approach. Their tests were performed on two synthetic WDNs. Obtained results verified the main assumptions of their proposed methodology. However, the main disadvantage of this method backs to the initial trust in the health of the WDNS, which is hard on large scale WDNs. The critical drawback of this work however, is the complexity of acquiring values of the required parameters that decreases the reliability of its results. Extensive reviews of recent advances on leak detection would be found in [1], [2], [22], [23].

We argue that any practical solution for leak detection should be able to detect multiple leaks in pipe networks and should identify in what pipe the leaks have happened. Also, it should be easily scalable and its deployment and maintenance costs should be reasonable. Toward this aim, in this paper, a novel model-based approach for leak detection in pipe networks is presented.

Transient-based methods, which rely on hydraulic equations, cannot solely be used in real applications. This is based on the fact that such methods cannot be easily extended to large systems. Noises in background can drastically affect their performance, and despite their complexity, they are not as effective as expected. In our proposed method, we benefit the use of hydraulic equations and the historical and real-time sensor data. We use hydrology relations to generate several features that are more sensitive for detecting leaks in pipe networks. In effect, by using hydraulic equations on one hand and setting the relation between flow data and daily demand in WDNs on the other hand, we find discriminating and more sensitive features like velocity for detecting and predicting leaks.

In order to test the effectiveness of the proposed method we use the data set that is gathered from a real WDN. The data set is used to simulate the WDN. The proposed methodology is then validated on different scenarios that are devised with the simulator.

It should be noted that the proposed method is not tested on an actual WDN. The obtained results show the superiority of our proposed method over other existing methods. We will demonstrate that our proposed method can not only detect and locate a single leak in a single pipe but also can detect and locate multiple leaks in a large WDN with an acceptable accuracy. Moreover, the proposed method is fast and demands for less computational power. Also it is revealed that the method can detect not only burst type leaks but also is able to detect and localize small background leaks. Hence, using the proposed method and relying on simple and cheap low power consumption sensors that have reasonable installation and maintenance expenses, we can expect a healthy WDN.

This paper is organized as follows. In section 2, we review the hydrology related equations and basic principles of hydraulics. In section 3, we discuss our proposed method. Section 4 is devoted to the experimental results and the comparison of the proposed method with other works. Section 5 concludes the paper.

2 Preliminaries

In this section, we review important hydraulic principles that we have benefitted in our proposed method for leak detection.

Hazen-Williams

The Hazen-Williams equation is an empirical equation that has long been used for calculating the friction loss in pipe network protection systems. The Hazen-Williams head loss in terms of flow rate expression is used to establish a relationship between flow rate of fluid and the head loss for steady state situation. The Hazen-Williams equation in SI unit is as follows,

$$V = 0.848\, C\, R_h^{0.63} S^{0.54} \tag{1}$$

where, V denotes the velocity, R_h indicates hydraulic radius, S shows the slope of the energy grade line and C is the Hazen-Williams coefficient. Notably, in Eq. (1), C specifies the pipes roughness and is not a function of Reynolds number, as in other pressure loss equations. The Hazen-Williams formula has the advantage of being simple [24]. In our work, we use this equation for calculating the velocity as a notable feature.

Darcy-Weisbach

Darcy–Weisbach equation is proposed for finding head-loss [25]. Like velocity, head loss has an important role in our proposed method for leak detection. Darcy-Weisbach equation in SI unit is,

$$h_f = F \times \left(\frac{L}{D}\right) \times \left(\frac{v^2}{2g}\right) \tag{2}$$

where h_f and F respectively denote head loss (m) and friction factor. L and D indicate length of pipe work (m), and inner diameter of pipe work (m) respectively. v shows the velocity of fluid (m/s) and g is the acceleration due to the gravity (m/s²).

Pressure drop

Pressure drop is the difference of the pressure in two points in a WDN. In hydraulics, pressure corresponds to density, elevation and gravity and is calculated as [25],

$$p = \rho g \Delta h_f \tag{3}$$

Pressure drop however is calculated as,

$$\Delta p = F \times \left(\frac{\rho L}{D}\right) \times \left(\frac{v^2}{2}\right) \tag{4}$$

where,

ρ = density (kg/m^3)
P = pressure (Pa)
h_f = head loss (m)
F = friction factor
L = length of pipe work (m)
D = inner diameter of pipe work (m)
v = velocity of fluid (m/s)
g = acceleration due to gravity (m/s²)

3 Proposed Method

Our proposed method for detecting leaks relies on both historical and real-time data —that are gathered from sensors— and other features that we generate through using hydraulic equations discussed in Sec. 2.
Fig. 1 depicts the schema of our proposed method that is consisted of 3 stages, namely Preprocessing, Modeling and Evaluation. We will describe each stage in details in the following.

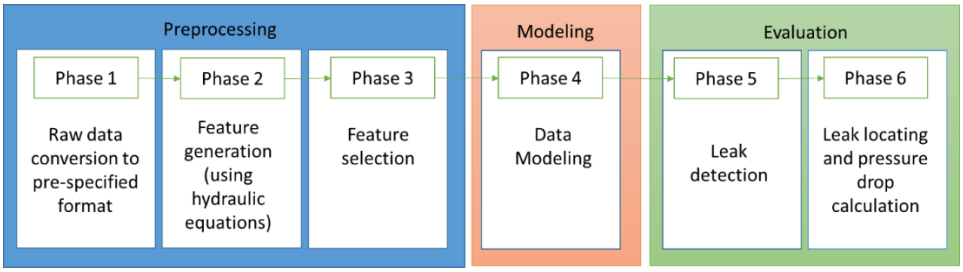

Fig. 1 Overview of the proposed method

3.1 Preprocessing

The aim of this stage is cleaning and putting the data in the format that is required by the next stages.

3.1.1 Raw data conversion

As per our method, the input data should include at the least, flow rate (liter per second) in each pipe, time stamp of the gathered data (collected from the flow sensors installed on WDN) and the WDN topology. Our proposed method requires a flow sensor be installed at each junction. Each junction and each pipe in the WDN should be identified with a unique id. By the topology of WDN, however we expect the junctions at each pipe, the length and diameter of the pipes and the elevation (position) of each junction be given.

Table 1 as an example, shows the topology of a WDN. In the table, Element and ID both denote pipes in the network. Scaled length is the length of the pipe, start and end nodes signifies the beginning and ending of the pipe with respect to their corresponding junctions. For instance, pipe P-2 connects the joints J-2 and J-3. The pipe P-2 is made up of ductile iron with the diameter of 350.8mm and its Hazen-William coefficient is 130. Junctions of the WDN shown in Table 1, however, are demonstrated in Table 2. At each junction a flow sensor is installed and its sensed data is collected in the junction's corresponding demand collection. In Table 2, Element and ID respectively represent a junction label and identifier. Elevation denotes the height at which the junction is installed.

An excerpt of flow rate that is recorded at junction J-2 is shown in the Figure 2. To reduce the volume of data, we define a time interval in which the median of the recorded flow rate will be calculated. This is due to the fact that sensors, record data very frequently which causes redundant data. More precisely, given Δt denote the time interval and assuming the data collection has started from time t_1, then for sensor J_x, the raw data will be divided into time windows of the length Δt. In each window the median of the flow rate will be calculated as the representative of the window and the time stamp based on Δt will be assigned to the median, i.e.,

$$median\left(f_{t_1}^{J_2}, f_{t_2}^{J_2}, \dots, f_{t_n}^{J_2}\right) = \bar{f}_{t_1}^{J_2} \quad (5)$$

$$median\left(f_{t_{n+1}}^{J_2}, f_{t_{n+2}}^{J_2}, \dots, f_{t_{n+n}}^{J_2}\right) = \bar{f}_{t_{1+\Delta t}}^{J_2} \quad (6)$$

$$\vdots$$

$$median\left(f_{t_{kn+1}}^{J_2}, f_{t_{kn+2}}^{J_2}, \dots, f_{t_{kn+n}}^{J_2}\right) = \bar{f}_{t_{1+n\Delta t}}^{J_2} \quad (7)$$

where $f_{t_n}^{J_2}$ denote the flow rate recorded at junction J_2 at the time t_n and $\bar{f}_{t_{1+n\Delta t}}^{J_2}$ is the median of the flow rate at junction J_2 in the time interval $\Delta t = [t_{kn+1}, t_{kn+n}], k \geq 0$. Δt can be identified experimentally based on the nature of the sensor and how frequent it collects data. We argue that reducing the flow rate as described, would also make the system tolerant to the missing values. That is, in the time interval of Δt, from the data generated by each flow sensor, only the median value of the flow rate is kept.

The output of this stage is the reduced flow rate at each junction considering the time interval Δt accompanied with the topology of the WDNs (Length of pipes, pipe's diameter, junction elevation, material and Hazen-William Coefficient). For instance, the flow rate in pipe P-1 measured at the junction R-1 (reservoir 1 where the flow is toward J-2) that would be delivered to the next stage is shown in Table 3. In this example, Δt is set to 1 hour.

Table 1 Topology of a sample WDN

Element	ID	Length (Scaled)(m)	Start Node	Stop Node	Diameter (mm)	Material	Hazen-Williams C
P-1	33	49	R-1	J-2	1000.80	Ductile Iron	130
P-2	35	12	J-2	J-3	350.80	Ductile Iron	130
P-8	44	16	J-4	J-3	255.58	Ductile Iron	130
P-10	48	17	J-7	J-8	310.48	Ductile Iron	130
P-11	50	4	J-8	J-9	35.48	Ductile Iron	130
P-12	52	17	J-9	J-10	210.48	Ductile Iron	130

Table 2 Junction description of the WDN described in Table 1.

Element	ID	Elevation(m)	Demand Collection
J-2	32	70	<Collection: 1 item>
J-3	34	69	<Collection: 1 item>
J-4	36	70	<Collection: 1 item>
J-7	45	0	<Collection: 1 item>
J-8	47	60	<Collection: 1 item>
J-9	49	50	<Collection: 1 item>
J-10	51	65	<Collection: 1 item>

Table 3 Reduced flow rate at R-1, calculated in the phase 1 of stage 1.

Element	Timestamp	Flow(l/s)
P-1	12:00:00 AM	1682.23
P-1	1:00:00 AM	1682.23
P-1	2:00:00 AM	4336.593
P-1	3:00:00 AM	5486.387
P-1	4:00:00 AM	7211.245
P-1	5:00:00 AM	7285.164
P-1	6:00:00 AM	7629.999

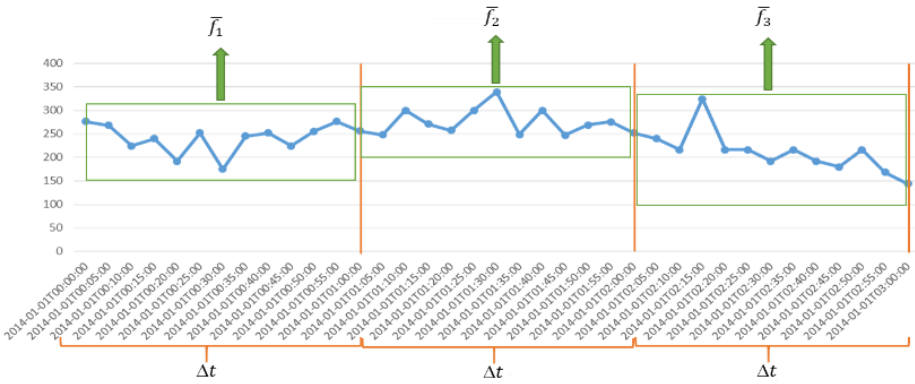

Fig. 2 Recorded flow rate data at Junction J-2 and its conversion to the reduced flow rate.

3.1.2 Feature generation

In this phase, we generate new features from raw data using hydraulic equations discussed in Sec 2. In subsequent sections we will show how influential these features are.

To generate velocity and head-loss, we use Eqs. (1) and (2) respectively. Calculating velocity based on Hazen-Williams Eq. (1), needs knowing the hydraulic radius, the slope of the energy grade line and Hazen-Williams coefficient - we expect WDN topology contains *C* for each pipe. To calculate Eq. (1), first we should calculate the slope of energy line as, [25]

$$S = \frac{hf}{L} \tag{8}$$

where *h*, *f* and *L* respectively denote head-loss, friction factor and length of the pipe. Hydraulic radius, however is calculated as, [25]

$$R_h = \frac{A}{P} \tag{9}$$

where *A* is the cross-sectional area and *P* is the wetted perimeter. For a pipe flowing full, hydraulic radius is equal to $R_h = D/4$ that *D* denotes the inner diameter of the pipe [25].

Head-loss, the other important feature we relies on, is generated by using the Darcy-Weisbach Eq. (2). To calculate the head-loss, friction factor should be found with reference to the moody diagram shown in Figure 3. To do so, initially Reynolds number and Relative Roughness should be calculated,

$$Re = \frac{VD}{v} \tag{10}$$

$$Relative\ roughness = \frac{e}{D} \tag{11}$$

where,
Re = Reynolds number
D = inner diameter (m)
V = velocity (m/s)
v = kinematic viscosity of the fluid (m^2/s).
e = pipe roughness (mm)

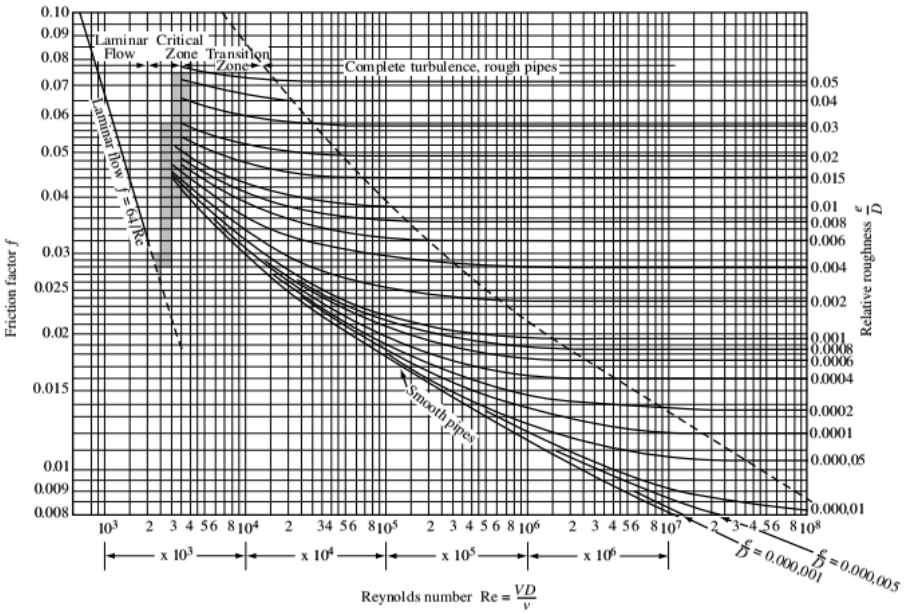

Fig. 3 The Moody diagram for the Darcy-Weisbach friction factor *f* [25]

We calculate the Relative roughness and Reynolds number to find the friction factor. For example, with the relative roughness of 0.004 and the Reynolds number equal to $3 * 10^4$ using the black line near 0.004 and moving on it to cross the vertical line at $3 * 10^4$ Reynolds number, the friction factor would be 0.03. Table 4 shows the results of applying this phase on the data shown in Table 3.

Table 4 Velocity and Headloss calculated for R-1 shown in Table 3

Element	Timestep	Flow(l/s)	Velocity (equation 1)	Headloss (equation 2)
P-1	12:00:00 AM	1682.23	0.021415	4.57281E-08
P-1	1:00:00 AM	1682.23	0.021415	4.57281E-08
P-1	2:00:00 AM	4336.593	0.055206	2.64145E-07
P-1	3:00:00 AM	5486.387	0.069844	7.52105E-07
P-1	4:00:00 AM	7211.245	0.091802	1.03465E-06
P-1	5:00:00 AM	7285.164	0.092743	1.13971E-06
P-1	6:00:00 AM	7629.999	0.097133	1.04528E-06

3.1.3 Feature selection

Feature selection is frequently used as a preprocessing step in machine learning. Feature sets sometimes contain several irrelevant and redundant features which impose negative effects on the performance of learning methods [26]. Feature selection in effect, is the process of choosing a subset of original features in order to optimally reduce the feature space according to certain evaluation criteria [27]. The heuristic methods that explore search space are commonly used for attribute subset selection. These methods are

typically greedy in nature [28]. Removing features with low variance, removes less effective data and reduces dataset dimension and the volume of sample sets. Therefore, setting a threshold for variance of each feature can be an approach for feature selection. All features whose variance does not meet a predefined threshold would be removed. By default, all zero-variance features should be removed, i.e. features that have the same value in all samples and consequently do not make any distinction.

As is shown in Figure 4, many features (including Velocity, Headloss) were generated based on flow rate and the water network specifications. However, by using the feature selection algorithm we keep more sensitive features. Velocity and Headloss are found as dominated features. The element (i.e. pipe), ID (Identifier of the pipe) and Length (Length of the pipe) are also kept from the topology.

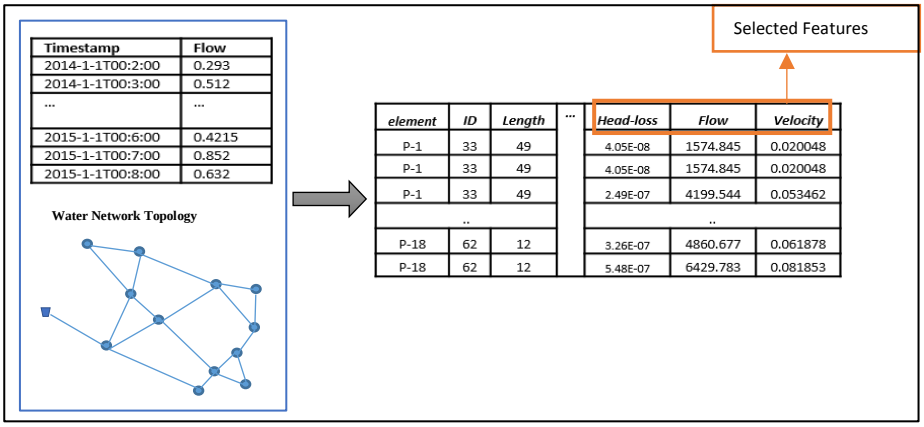

Fig. 4 Feature generation and feature selection

Since leak detection in water network is an imbalanced problem, we should use predictors that are based on classification models. The imbalanced problems are highly related to cost-sensitive learning. The costs of error for each class are not equal in imbalanced problems [29]. Therefore, choosing a suitable model and appropriate evaluation metrics can help making good decisions that would result in time and cost reduction.

We use different classifiers, namely, K-nearest-neighborhood (KNN), Decision Tree, Random Forest (ensemble method) and Bayesian. KNN is one of the most popular classifiers which is a supervised learning algorithm. The KNN classification algorithm predicts the test data's category according to the K training samples which are the nearest neighbors to the test sample, and embeds it in the category which has the largest category probability [30]. Decision trees are constructed in top-down recursive divide-and-conquer manner. They start with a training set of tuples and their associated class labels. The training data are recursively partitioned into smaller subsets as the tree is being built [29]. An ensemble for classification is a composite model, made up of a combination of classifiers. An individual classifier vote and a class label prediction is returned by the ensemble based on the collection of votes. Random Forest classifier is an ensemble of decision trees that let each of the trees vote for a class label. Bayesian classifiers are statistical classifiers that can predict class membership probabilities such as the probability that a given tuple belongs to a particular class. Each of these classifiers has pros and cons that are discussed in Table 5.

Table 5 Comparison of classifiers KNN, Decision Tree, Random Forest and Bayesian Network [31]-[32]

	Pros	**Cons**
KNN	Robust when using small k Easy to implement	Each test data may be close to many points,
Random Forest	Improved predictive performance,	Hard to analyze output
Decision tree	Data classification without much calculations,	Tree splitting is locally greedy,
Bayesian	Fast and efficient computation, Quick training	Impractical for data sets with many features

The test data should be prepared and delivered to each classifier based on the classifier's requirements. The performance of this phase is determined by comparing its predicted classes with the actual classes. To make these models more practical and useable, their parameters should be tuned based on the quality of results and measures. To be more precise, the parameters we have considered for the mentioned models are discussed in the following.

- Decision Tree:
  - criterion: criteria are "gini" for the Gini impurity and "entropy" for the information gain.
  - splitter: Supported strategies are "best" to choose the best split not "random" from the best chooses.
- Random forest classifier:
  - n_estimators: the number of trees in the forest (n = 100)
- k-nearest neighbors:
  - n_neighbors: Number of neighbors to use (n = 5)
- Gaussian Naive Bayes:
  - priors: Prior probabilities of the classes (none)

When the model is trained it can detect in which pipe a leak has happened. An example is shown in Fig. 5. Test data depicted in Fig.5(A) shows if pipes have had a leakage (Target=1) or not (Target=0). Junctions corresponding to the pipe are identified by knowing the WDN topology through application of the "finding junctions" module - Fig. 5(B). Relying on the WDN topology and the corresponding sensor data, then we calculate the pressure drops at the leaks by using equation 2 and 3 -Fig. 5(C-D).

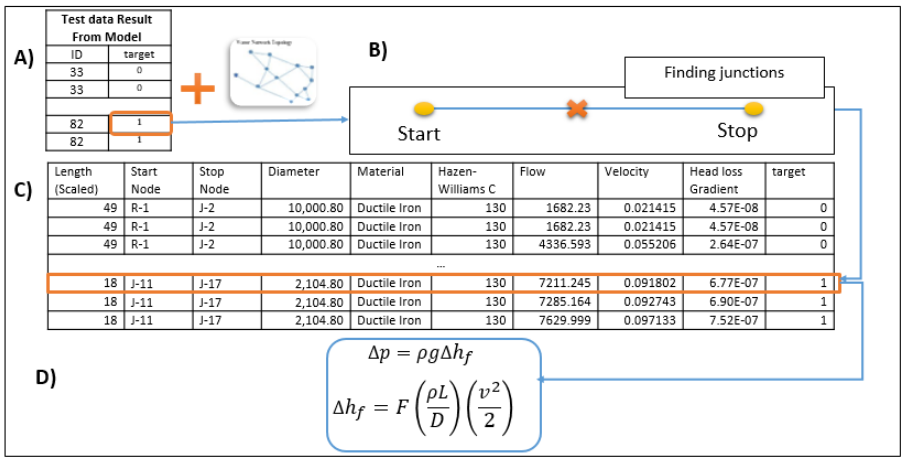

Fig. 5 Building leaking model, (A) preparing test data containing leaking and healthy pipes (B) Identifying junctions associated with pipes , (C), (D) Determining pressure drop at the leakages

5 Experiments

In order to test the proposed method, we have relied on a real data set. However, since the data set was offline it was not possible to test all plausible leaks. Hence, initially we simulated the WDN using Water GEMS simulator. The simulator modeled the WDN and then we introduced different leak scenarios in the WDN. In the following we describe the used dataset and the simulator and then will discuss different scenarios that we have developed for testing the proposed method.

5.1 Data set

We use the Vitens company data set of the open challenge 2015 that describes the water distribution networks of Leeuwarden city in Netherland [33]. This data set contains data of flow, pressure, temperature, turbid, conductivity and acidity. We transfer the cubic meter per hour unit of flow to liter per second. In our proposed method we do not use temperature, turbid, conductivity and acidity features since the number of the sensors that were used in Leeuwarden water network for collecting such information were few. Moreover, installing such sensors in any WDN imposes high costs. Our proposed method is aimed to provide the highest accuracy for leak detection with the least cost.

Each sensor data in the dataset is accompanied by a timestamp that shows the time the data item is gathered. Moreover, there are some comments in the dataset that gives information about location, engineering units and sensor accuracy. A sample record from the dataset of Vitens Company is shown in Figure 6. Comments in the file of data set contain valuable information that help identifying the types of sensors, exact location of sensors, engineering units, start and ending date of collecting data and finally the time resolution.

# PI Point:	FR-MADB-FT01-meetwaarde				
# Description:	1) Gorredijk -> Grou (Easterboarn 2 8495 NB Aldeboarn)				
#	2)				
# Engineering units:	m3/h				
# Accuracy:	Â± (1 m3/h				
# Stepped data:	False				
# Data dumped at:	30-11-2015 19:16:23				
# Time resolution limit: 60s					
# START AT 2014-01-01T00:00:00					
Timestamp	Value				
2014-01-01T00:00:00	1.025641				
2014-01-01T00:01:00	0.8791209				
2014-01-01T00:02:00	0.1953602				
2014-01-01T00:03:00	1.269841				
2014-01-01T00:04:00	0.2930403				
2014-01-01T00:05:00	0.6837607				
2014-01-01T00:06:00	0.3907204				
2014-01-01T00:08:00	0.6349207				

Fig. 6 The sample of recorded data from Vitens

5.2 Simulator

One of the general, comprehensive and simple water distribution modeling applications is Water GEMS [34]. For modeling a water network with Water GEMS first pipes, junctions and reservoirs should be initialized by using tools in the application and information from dataset. Then inputs like demand and the pipe materials should be provided.

In order to reconstruct the pipeline network, first we located the junctions and sensors on the google map using the information provided in the dataset. Fig.7 shows locations of the sensors on the pipeline network of the Leeuwarden city. Using the Water GEMS functionality the map was put as a background layer for drawing the topology. Fig.8 shows the abstract map of the WDN that is generated by the simulator. As can be seen in Fig. 8 this network consists of one reservoir, 27 pipes and 18 junctions. It should be noticed that a flow sensor is installed on each junction.

Fig. 7 sensor location in Leeuwarden city in Netherland

We set the Water GEMS parameters related to the junctions and pipes using the data read from the dataset as,

Junction: General (ID, label), Geometry (x, y), Demand (Demand Collection), Physical (Elevation, Zone), Water Quality, Active Topology (True, False)

Pipe: General (ID, label), Geometry (collection), initial setting (open/close), Physical (Elevation, Diameter, Material, Hazen-Williams C, Length)

Feeding Water GEMS with the data read from the dataset, the WDN was simulated. Figure 9 for instance, shows the flow in a typical day at the junction J-2 generated by the simulator.

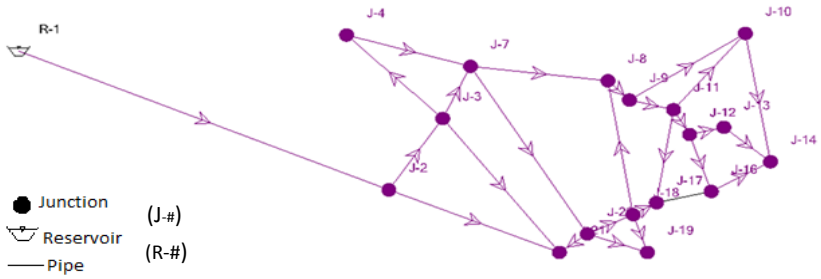

Fig. 8 water network topology

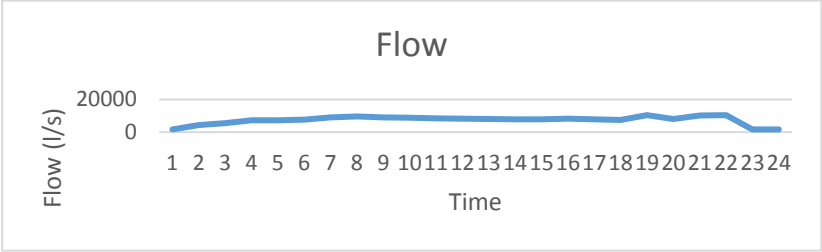

Fig. 9 24 hour Flow pattern in j-2

5.3 Scenarios

In order to test the performance of the proposed method in this paper, we have devised 8 different leak scenarios that depict most possible leaks in a WDN. As the WDN is in the residential area, any spot in the WDN can be a leak location and all pipes can be assumed as a damaged pipe. In these 8 scenarios, we have considered all the different situations and have investigated them. For instance, we have devised different scenarios with long and short length pipes, different pressure (burst and slight leaks) or various leak distance from the sensors. Scenarios are shown in Table 7. In order to show how effective the proposed method is, for each scenario we calculate the accuracy, precision and f-score. In general, accuracy tells the percentage of data that are correctly classified by our model. Therefore, it would be a proper metric when classes are evenly distributed. However, since our data is imbalanced, we calculate precision to show its exactness. F-Score, as the harmonic mean of precision and recall are also reported. Table 6 shows the confusion

matrix and accuracy, precision, recall and F-Score are calculated using equations (12)-(15). Obtained results for each scenario is depicted in the last column of Table 7.

Table 6 Confusion Matrix [29]

		Predicted class	
		Yes	No
Actual class	Yes	TP	FN
	No	FP	TN

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}} \tag{12}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{13}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{14}$$

$$\text{F score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{percision} + \text{recall}} \tag{15}$$

Table 7 Scenario Description and results of applying the proposed method

(a)

			Number of Leaks	Leaking pipe	junctions	pressure (Kpa)	Demand Flow (L/S)	Results
scenario-1	Train Data	Day-1	1	P-25	(j-18, j-20)	1	1	
	Test Data	Day-2	1	P-10	(j-7, j-8)	1	1	

Model	**accuracy**	**Precision**	**recall**	**f score**
RF	93.4	100	92.9	96.3
Dtree	91.7	100	91.07	95.3
KNN	91.7	93.02	98.4	95.6
Bayes	95.8	100	95.2	97.5

(b)

			Number of Leaks	Leaking pipe	junctions	pressure (Kpa)	Demand Flow (L/S)	Results
Scenario-2	Train Data	Day-1	3	p-33	(j-4, j-7)	200	200	
				p-13	(j-10, j-11)	200	200	
				p-21	(j-17,j16)	200	200	
	Test Data	Day-2	1	p-35	(j-7, j-20)	200	200	

Model	**accuracy**	**Precision**	**recall**	**f score**
RF	94.7	100	93.92	96.8
Dtree	93.6	100	92.6	96.1
KNN	92.27	97	93.9	95.4
Bayes	96.13	100	95.5	97.7

(c)

			Number of Leaks	Leaking pipe	junctions	pressure (Kpa)	Demand Flow (L/S)	Results
Scenario-3	Train Data	Day-1	4	P-10	(j-7,j-8)	1	1	
				P-2	(j-2,j-3)	2	2	
				P-32	(j-12,j-16)	1	1	
				p-12	(j-9,j-10)	1	1	
	Test Data	Day-2	2	P-31	(j-17,j-11)	1	1	
				P-17	(j-10,j-14)	1	1	

Model	**accuracy**	**Precision**	**recall**	**f score**
RF	94.2	100	93.2	96.5
Dtree	91.58	100	90.24	94.8
KNN	89.6	94.07	93.92	94
Bayes	96.2	100	95.6	97.7

(d)

			Number of Leaks	Leaking pipe	junctions	pressure (Kpa)	Demand Flow (L/S)	Results
Scenario-4	Train Data	Day-1	4	P-10	(j-7,j-8)	1	1	
				P-2	(j-2,j-3)	1	1	
				P-32	(j-12,j-16)	1	1	
				P-12	(j-9,j-10)	1	1	
	Test Data	Day-2	6	P-35	(j-7,j-20)	1	1	
				P-24	(j-20,j-19)	1	1	
				P-14	(j-11,j-12)	1	1	
				P-20	(j-16,j-14)	1	1	
				P-8	(j-4,j-3)	1	1	
				P-26	(j-2,j-21)	1	1	

Model	**accuracy**	**Precision**	**recall**	**f score**
RF	96.4	100	94.4	97.4
Dtree	94.3	100	91.04	95.3
KNN	78.18	76.99	93.7	84.5
Bayes	96.9	100	95.2	97.5

(e)

			Number of Leaks	Leaking pipe	junctions	pressure (Kpa)	Demand Flow (L/S)	Results
Scenario-5	Train Data	Day-1	1	P-33	(j-4,j-7)	1	1	
	Test Data	Day-2	6	P-35	(j-7,j-20)	1	1	
				P-24	(j-20,j-19)	1	1	
				P-14	(j-11,j-12)	1	1	
				P-20	(j-16,j-14)	1	1	
				P-8	(j-4,j-3)	1	1	
				P-26	(j-2,j-21)	1	1	

Model	**accuracy**	**Precision**	**recall**	**f score**
RF	99.15	100	98.6	99.3
Dtree	94.3	100	94.8	97.3
KNN	76.6	73.18	99.8	84.4
Bayes	96.4	94.7	100	97.3

(f)

			Number of Leaks	Leaking pipe	junctions	pressure (Kpa)	Demand Flow (L/S)	Results
Scenario-6	Train Data	Day-1	1	P-33	(j-4,j-7)	1	1	
	Test Data	Day-2	4	P-35	(j-7,j-20)	50	35	
				P-24	(j-20,j-19)	90	70	
				P-14	(j-11,j-12)	70	45	
				P-20	(j-16,j-14)	60	40	

Model	**accuracy**	**Precision**	**recall**	**f score**
RF	95.3	100	93.7	96.7
Dtree	88.3	100	84.3	91.5
KNN	76.9	76.61	99.1	86.4
Bayes	89.9	100	86.4	92.7

(g)

			Number of Leaks	Leaking pipe	junctions	pressure (Kpa)	Demand Flow (L/S)	Results
Scenario-7	Train Data	Day-1	4	P-35	(j-7,j-20)	50	35	
				P-24	(j-20,j-19)	90	70	
				P-14	(j-11,j-12)	70	45	
				P-20	(j-16,j-14)	60	40	
	Test Data	Day-2	1	P-33	(j-4,j-7)	1	1	

Model	**accuracy**	**Precision**	**recall**	**f score**
RF	96.14	100	95.8	97.8
Dtree	86.7	100	85.69	92.2
KNN	92.7	94.9	97.3	96.1
Bayes	100	100	100	100

(h)

			Number of Leaks	Leaking pipe	junctions	pressure (Kpa)	Demand Flow (L/S)	Results
Scenario-8	Train Data	Day-1	No leak					
		Day-2	No leak					
		Day-3	1	P-31	(j-17,J-11)	1	1	
		Day-4	2	P-17	(J-10,j-14)	1	1	
				P-30	(J-18,J-8)	1	1	
	Test Data	Day-5	6	P-35	(j-7,j-20)	1	1	
				P-24	(j-20,j-19)	1	1	
				P-14	(j-11,j-12)	1	1	
				P-20	(j-16,j-14)	1	1	
				P-8	(j-4,j-3)	1	1	
				P-26	(j-2,j-21)	1	1	

Model	**accuracy**	**Precision**	**recall**	**f score**
RF	72	71.5	95.2	81.6
Dtree	72.3	71.87	92.9	81
KNN	79.1	75.46	99.6	85.8
Bayes	79.1	75.46	99.6	85.8

To implement scenario-1 shown in Table 7(a), we intentionally applied one leak in the pipe (P-25) that is between the junctions (j-18, j-20) with the pressure of 1 Kpa and demand leak of 1 L/s. The pressure, represents the pressure at the leak and Demand Flow indicates the waste of water in the unit of Liter per second. After building the model, we generated the test data by adding a leak in the pipe (P-10) between junctions (j-7, j-8) with the same pressure of 1 Kpa and demand leak of 1 L/s. The results are shown in Table 7(a). The proposed method has been able to identify the leak one hour after the leakage is initiated. However, the evaluation of the measurements are based on the data gathered for the whole 27 pipes in 24 hours. In scenario-1 with all models, accuracy is the lowest

measure and f-score is the highest. In terms of f-score, which is the combination of recall and accuracy, performance of Bayesian and KNN are equal and slightly higher than the other models. As we mentioned, F-score is the proper measure for this kind of problems.

In scenario-2 there are 3 leaks in train data and 1 leak in test data. In scenario-3 we have devised 4 leaks in different parts of the city in train data but in the test data, two leaks are assumed. In scenario-4, the number of leaks in train data is 4 but there are 6 leaks in test data. In scenario-5 the number of leaks in train data is 1 whereas there are 6 leaks in test data. Scenarios 6 and 7 have different leaks in train and test data with varying pressures and demand flows. In scenario-8 we considered data of 4 days for training. Results obtained for each scenario is shown in Table 7.

Figures 10-13 summarizes the accuracy, precision, recall and f-measures calculated for all the scenarios. Accuracy shows how well the classifier can find the leaks. As can be seen in Fig. 10 most of the accuracy measures are high which is due to the imbalanced problem. Sensitivity (recall) and precision are more reasonable measures for evaluation of the performance of the proposed method.

From Fig. 11 it can be observed that random forest and decision tree are more reliable and KNN is good in the early scenarios that have more leaks in train data than test data. Although high precision is good but is not enough since if we detect a leak that actually has not happened, its associated costs would be high, because of digging the earth and time and effort that should be consumed. Recall or sensitivity depicts the leaks that are detected among all the actual leaks. If a leak happens that the system cannot find it, financial costs and environmental damages can be important issues to be considered. Therefor recall would be an appropriate measure.

All the obtained f-scores are shown in Fig. 13. In this figure, it is noticeable that 80.5% of the models are getting results above 92 percent in all scenarios. The best result achieved in most of the scenarios is by Naïve Bayesian Model that is in [85.8,100]. Random Forest as an ensemble shows to be a reliable model. Decision Tree remains steady in all the scenarios. However, the KNN decreases in scenarios 5, 6 and 7, which means performance of this model directly depended on the number of leaks in train data. All the models for scenario 8 have provided poor results that can be explained due to the poor or inadequate training data. Pros and cons of the used classifiers are summarized in Table 9.

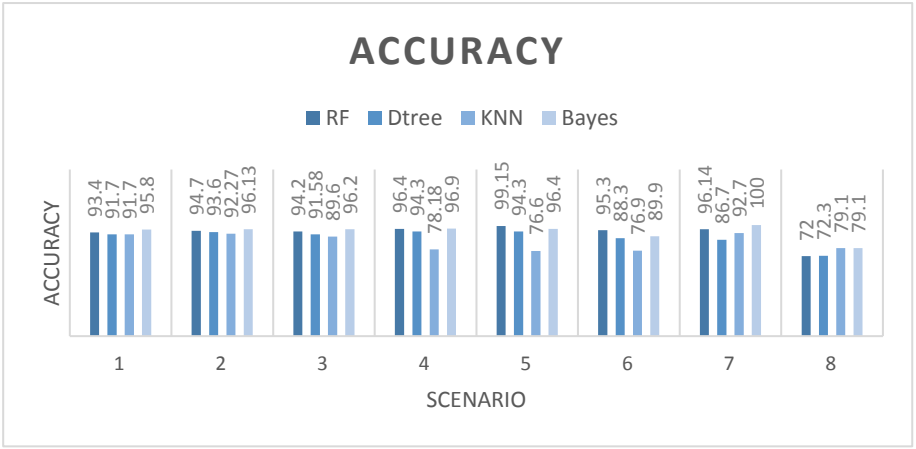

Fig.10 Accuracy of leak detection of all scenarios

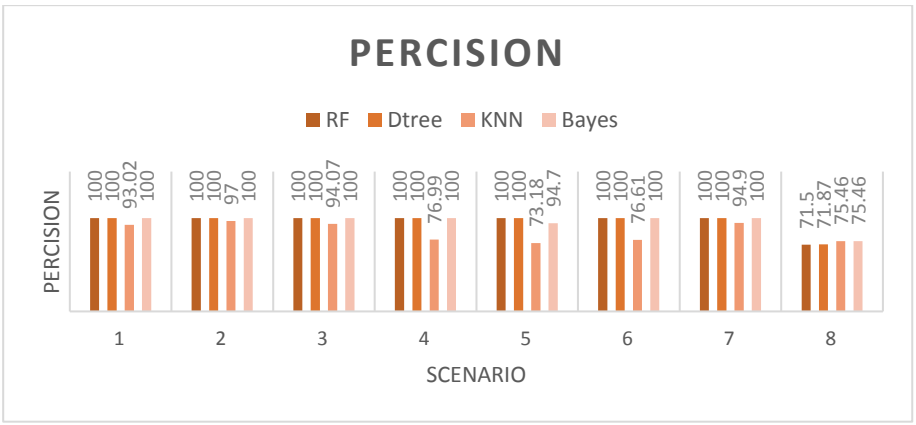

Fig.11 Leak detection Precision of all scenarios

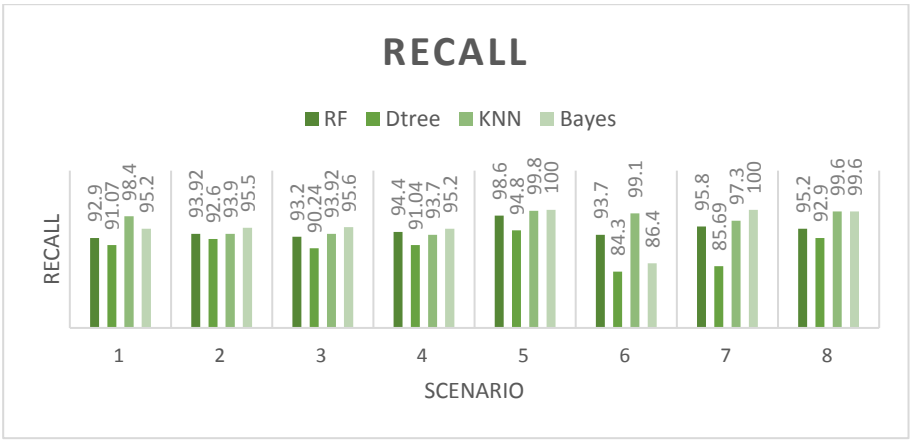

Fig.12 Leak detection Recall of all scenarios

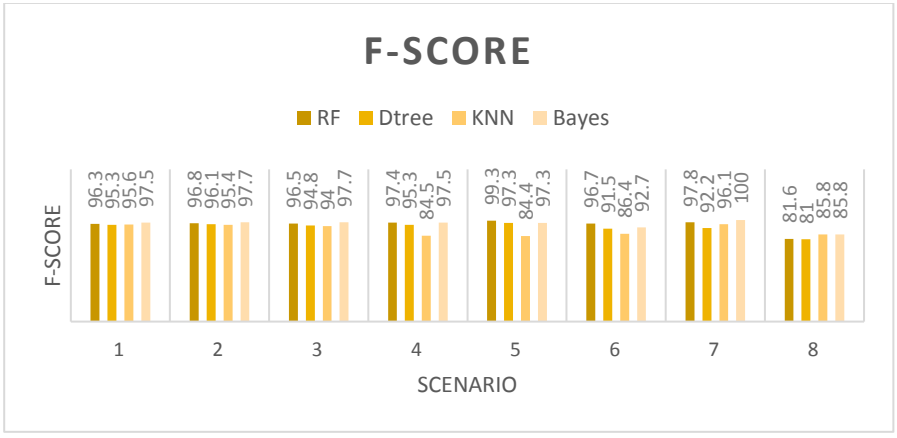

Fig.13 Leak detection F-score of all scenarios

The average of all measures for all the scenarios for each model is shown in Table 8. As can be seen in Table 8 the highest f-score is achieved by using Bayesian model and in general Bayesian model has outperformed other models.

Table 8 Average of all measures for all scenarios for each model

	RF	**DTree**	**KNN**	**Bays**
precision	96.4375	96.48375	85.15375	96.27
Recall	94.715	90.33	96.965	95.9375
f-score	95.3	92.9375	90.275	95.775
accuracy	92.66125	89.0975	84.63125	93.80375

Table 9 Comparison of classifiers KNN, Decision Tree, Random Forest and Bayesian Network With respect to leak detection problem in WDNs

	Pros	**Cons**
KNN	More accurate with the imbalanced data	Under average performance with balanced data
Random Forest	robust and trustable in different circumstances	Degraded performance with imbalanced data.
Decision tree	Good precision	Poor Recall
Bayesian	Effective and robust	-

5.4 Comparison

We have compared our proposed method with Soldevila *et al.*[18] and Buchberger *et al*. [20]. In Soldevila *et al*. [18] method, leaks in Water Distribution Networks (WDNs) are detected based on classifiers (KNN, Bayesian) and pressure models. Table 10 shows the results obtained by applying the method of [18] on our preprocessed data of scenarios described in Table 7. It shows that our proposed method outperforms the method of [18]. The performance of our method is better which is due to the preprocessing and effective feature generation.

Table 10 Average accuracy of [18] and proposed method on the scenarios of Table 7.

Proposed Method-KNN (accuracy)	**Proposed Method-Bayesian (accuracy)**	**Soldevila-KNN [13] (Accuracy)**	**Soldevila-Bayesian[13] (Accuracy)**
84.63125	93.80375	74.19	83.87

Buchberger *et al.* in [20] have presented a leak detection algorithm that uses statistical computations and analysis on the sequence of flow as the data set is truncated progressively. The similarity of our proposed method and their method is that both methods are using Flow as the main feature. As discussed before, Buchberger *et al.*'s method has many limitations. Notably it can only detect if leak has happened in the WDN but cannot recognize the number and placement of the leaks. Table 11 compares the results of our proposed method with [20] on all the defined scenarios. As can be seen, our proposed method is faster and more accurate. Also we have been able to locate the leaks and determine the pressure at each leak. Both algorithms were run on the computer with Intel Core i5, 2.30 GHz CPU, and RAM 4.00 GB. As a comprehensive comparison Table 12 reports the difference and similarity of our proposed method with Soldevila *et al*.[18] and Buchberger *et al*.[20]

Table 11 Comparing our method and [20] for the 8 scenarios (QL in the Buchberger *et al*.[18] denotes Flow(L/s) and P in our method stands for Pressure (Kp))

Scenario	Leaks		Our proposed method	Buchberger *et al*.'s method [18]
1	1 leak for Test		Real time, 1 leak detected Run time: 6.99699 sec. Leaking Pipe: (J-7, J-8) Leak P: 0.3276Kp	1 leak detected QL == 361.2159 Run time: 263.23099 sec.
	P:1 Kpa	F:1 L/s		
2	1 leak for Test		Real time, 1 leak detected Run time: 8.375 sec. Leaking Pipe: (J-7, J-20) Leak P: 208.9875 Kp	1 leak detected QL == 586.45 Run time:162.585000038 sec.
	P2:200 Kpa	F2:200 L/s		
3	2 leak for Test		Real time, 2 leaks detected f-score = 97.7 Run time: 7.49099993706 sec. Leaking Pipes: (J-14, J-10); (J-11, J-17) Leak P1: 2.63777 KP Leak P2: 1.3517 KP	1 leak detected QL =1533.37 Run time: 145.220000029 sec.
	P1:1 Kpa P2:1 Kpa	F1:1 L/s F2:1 L/s		
4	6 leak for Test		Real time, 6 leaks detected f-score = 97.5 Run time: 7.52699995041 sec. Leaking Pipes: (J-19, J-20), (J-14, J-16), (J-11, J-12), (J-7, J-20), (J-21, J-2), (J-4, J-3) Leak P1: 0.0905 KP Leak P2: 2.1366 KP Leak P3: 2.4376 KP Leak P4: 2.1814 KP Leak P5: 0.14805 KP Leak P6: 0.20726 KP	1 leak detected QL = 1529.464 Run time: 146.371999979 sec.
	P1:1 Kp P2:1 Kpa P3:1 Kpa P4 :1 Kpa P5:1 Kpa P6:1 Kpa	F1:1 L/s F2:1 L/s F3:1 L/s F4:1 L/s F5:1 L/s F6:1 L/s		
	6 leak for Test		Real time,	1 leak detected

5	P1:1 Kpa P2:1 Kpa P3:1 Kpa P4 :1 Kpa P5:1 Kpa P6:1 Kpa	F1:1 L/s F2:1 L/s F3:1 L/s F4:1 L/s F5:1 L/s F6:1 L/s	6 leaks detected f-score = 97.3 Run time: 8.43400001526 sec. Leaking Pipe: (J-19, J-20), (J-14, J-16), (J-11, J-12) ,(J-7, J-20), (J-21, J-2), (J-4, J-3) Leak P1: 0.2584 Kp Leak P2: 0.0844 Kp Leak P3: 0.2265 Kp Leak P4: 23.49785 Kp Leak P5: 0.1223 Kp Leak P6: 0.2072 Kp	QL = 1497.040 Run time: 139.127000093 sec
6	4 leak for Test		Real time, 4 leaks detected f-score = 92.7 Run time: 7.64899992943 sec. Leaking Pipe: (J-19, J-20), (J-14, J-16), (J-11, J-12), (J-7, J-20) Leak P1: 2.0746 Kp Leak P2: 33.8544 Kp Leak P3: 59.3092 Kp Leak P4: 13.8771 Kp	1 leak detected QL = 1536.6783 Run time: 139.628000021 sec.
	P1:50 Kpa P2:90 Kpa P3:70 Kpa	F1:35 L/s F2:70 L/s F3:45 L/s		
	P4:60 Kpa	F4:40 L/s		
7	1 leak for Test		Real time, 1 leak detected f-score = 100 Run time: 6.48099994659 sec. Pipe: (J-7, J-4) Leak P1: 1.39988 Kp	1 leak detected QL = 496.592 Run time: 128.111000061 sec.
	P1:1 Kpa	F1:1 L/s		
8	6 Test leaks		Real time, 6 leaks detected Run time: 14.87 sec. f-score = 87.6 Pipe: (J-19, J-20), (J-14, J-16), (J-11,J-12), (J-7, J-20), (J-21, J-2), (J-4, J-3) Leak P1: 0.1410 Kp Leak P2: 1.2866 Kp Leak P3: 4.0538 Kp Leak P4: 2.7240 Kp Leak P5: 10.166 Kp Leak P6: 0.3276 Kp	1 leak detected QL = 816.12 Run time:272.248 Sec.
	P1:1 Kpa P2:1 Kpa P3:1 Kpa P4 :1 Kpa P5:1 Kpa P6:1 Kpa	F1:1 L/s F2:1 L/s F3:1 L/s F4:1 L/s F5:1 L/s F6:1 L/s		

Table 12 Comparing proposed method with [20] and [16]

Method	One leak detection	Multiple leaks detection	Leak prediction	Run time	Leak Location
Our proposed method	✓	✓	✓	Low	✓
Buchberger *et al.*[20]	✓	-	-	high	-
Soldevila *et al.*[18]	✓	✓	✓	Low	-

6 Conclusion

Leak detection in the pipeline network is a vital issue not only from financial perspective but also for the damages that can be imposed on the environment. There are many different methods to detect leaks that most of them have focused on detecting burst-type leakages[1]. In this paper we proposed a novel hybrid method that is transient-based and model-based for detecting leakages. The proposed method relies on historical data as well as real-time data that are gathered by flow sensors that are installed on the junction of pipeline network. By processing the gathered data and using hydraulics equations, new features of velocity and head loss are generated. By means of various classification algorithms we showed that our proposed algorithm is able to detect and locate single and multiple leaks in different pipes of a WDN with an acceptable f-scores. The proposed method is tested under different scenarios and leak conditions. The results show the superiority of this method over other methods in different aspects such as cost, time, leak locating and reliability. Due to the leakages applied -with different pressures and demand flows- in the tested scenarios, the proposed method shows its applicability in both burst-type and background type leakages. Due to the required sensors, maintenance of the system is easy and low-cost, however, the proposed method, to function appropriately should be provided with the topology of WDN, pipes' sizes and pipes' materials which can be a burdensome. Moreover, the detection of leakages on the junction can be a limitation for our proposed method that is the subject of future works.

References

[1] K. B. Adedeji, Y. Hamam, B. T. Abe, and A. M. Abu-Mahfouz, "Towards Achieving a Reliable Leakage Detection and Localization Algorithm for Application in Water Piping Networks: An Overview," *IEEE Access*, vol. 5, pp. 20272–20285, 2017, doi: 10.1109/ACCESS.2017.2752802.

[2] H. Ali and J. H. Choi, "A review of underground pipeline leakage and sinkhole monitoring methods based on wireless sensor networking," *Sustain.*, vol. 11, no. 15, 2019, doi: 10.3390/su11154007.

[3] H. Ali and J. Choi, "Risk Prediction of Sinkhole Occurrence for Different Subsurface Soil Profiles due to Leakage from Underground Sewer and Water Pipelines," *Sustainability*, vol. 12, no. 1, p. 310, 2019, doi: 10.3390/su12010310.

[4] S. K. Sinha and M. A. Knight, "Intelligent system for condition monitoring of underground pipelines," *Comput. Civ. Infrastruct. Eng.*, vol. 19, no. 1, pp. 42–53, 2004, doi: 10.1111/j.1467-8667.2004.00336.x.

[5] A. Sadeghioon, N. Metje, D. Chapman, and C. Anthony, "SmartPipes: Smart Wireless Sensor Networks for Leak Detection in Water Pipelines," *J. Sens. Actuator Networks*, vol. 3, no. 1, pp. 64–78, 2014, doi: 10.3390/jsan3010064.

[6] B. Van Hieu, S. Choi, Y. U. Kim, Y. Park, and T. Jeong, "Wireless transmission of acoustic emission signals for real-time monitoring of leakage in underground pipes," *KSCE J. Civ. Eng.*, vol. 15, no. 5, pp. 805–812, 2011, doi: 10.1007/s12205-011-0899-0.

[7] Y. A. Khulief, A. Khalifa, R. Ben Mansour, and M. A. Habib, "Acoustic Detection of Leaks in Water Pipelines Using Measurements inside Pipe," *J. Pipeline Syst. Eng. Pract.*, vol. 3, no. 2, pp. 47–54, 2012, doi: 10.1061/(ASCE)PS.1949-1204.0000089.

[8] S. C. Huang, W. W. Lin, M. T. Tsai, and M. H. Chen, "Fiber optic in-line distributed sensor for detection and localization of the pipeline leaks," *Sensors Actuators, A Phys.*, vol. 135, no. 2, pp. 570–579, 2007, doi: 10.1016/j.sna.2006.10.010.

[9] O. Hunaidi and P. Giamou, "Ground-Penetrating Radar For Detection Of Leaks In Buried Plastic Water Distribution Pipes," no. May, pp. 27–30, 1998.

[10] R. A. Cody, B. A. Tolson, and J. Orchard, “Detecting Leaks in Water Distribution Pipes Using a Deep Autoencoder and Hydroacoustic Spectrograms,” *J. Comput. Civ. Eng.*, vol. 34, no. 2, pp. 1–8, 2020, doi: 10.1061/(ASCE)CP.1943-5487.0000881.

[11] R. J. Cintra, T. de Oliveira, and M. P. Mintchev, “Leakage Prevention and Real-Time Internal Detection in Pipelines Using a Built-In Wireless Information and Communication Network,” *SPE J.*, no. March 2019, pp. 1–12, 2020, doi: 10.2118/201096-pa.

[12] Y. Liu, X. Ma, Y. Li, Y. Tie, Y. Zhang, and J. Gao, “Water pipeline leakage detection based on machine learning and wireless sensor networks,” *Sensors (Switzerland)*, vol. 19, no. 23, pp. 1–21, 2019, doi: 10.3390/s19235086.

[13] A. Al-Khomairi, “Leak detection in long pipelines using the least squares method Leak detection in long pipelines using the least squares method Détection de fuite dans de longues canalisations en utilisant la méthode des moindres carrés,” *J. Hydraul. Res.*, vol. 463, no. 3, pp. 392–401, 2008, doi: 10.3826/jhr.2008.3191.

[14] P. Taylor, W. Mpesha, M. H. Chaudhry, and S. L. Gassman, “Leak detection in pipes by frequency response method using a step excitation Leak detection in pipes by frequency response method using a step excitation Detection des fuites de tuyauteries par une methode de réponse en frequence utilisant un échelon d ’,” *J. Hydraul. Res.*, no. April 2013, pp. 37–41, 2010.

[15] Y. Asada, M. Kimura, I. Azechi, T. Iida, and N. Kubo, “Leak detection by monitoring pressure to preserve integrity of agricultural pipe,” *Paddy Water Environ.*, vol. 17, no. 3, pp. 351–358, 2019, doi: 10.1007/s10333-019-00730-5.

[16] P. Cuguero-Escofet, J. Blesa, R. Perez, M. Cuguer??-Escofet, and G. Sanz, “Assessment of a leak localization algorithm in water networks under demand uncertainty,” *IFAC-PapersOnLine*, vol. 28, no. 21, pp. 226–231, 2015, doi: 10.1016/j.ifacol.2015.09.532.

[17] A. Soldevila, R. M. Fernandez-Canti, J. Blesa, S. Tornil-Sin, and V. Puig, “Leak localization in water distribution networks using Bayesian classifiers,” *J. Process Control*, vol. 55, pp. 1–9, 2017, doi: 10.1016/j.jprocont.2017.03.015.

[18] A. Soldevila, S. Tornil-sin, J. Blesa, M. Rosa, and V. Puig, “Modeling and Monitoring of Pipelines and Networks,” *Springer Int. Publ. AG 2017 C. Verde L. Torres (eds.), Model. Monit. Pipelines Networks, Appl. Cond. Monit.*, vol. 7, 2017, doi: 10.1007/978-3-319-55944-5.

[19] J. Mashford, D. De Silva, D. Marney, and S. Burn, “An approach to leak detection in pipe networks using analysis of monitored pressure by support vector machine,” *Third Int. Conf. Netw. Syst. Secur.*, no. Figure 1, pp. 534–539, 2009, doi: 10.1109/NSS.2009.38.

[20] S. G. Buchberger and G. Nadimpalli, “Leak Estimation in Water Distribution Systems by Statistical Analysis of Flow Readings,” *J. Water Resour. Plan. Manag.*, vol. 130, no. 4, pp. 321–329, 2004, doi: 10.1061/(ASCE)0733-9496(2004)130:4(321).

[21] G. Mazzolani, L. Berardi, D. Laucelli, A. Simone, R. Martino, and O. Giustolisi, “Estimating Leakages in Water Distribution Networks Based Only on Inlet Flow Data,” *J. Water Resour. Plan. Manag.*, pp. 1–11, 2017, doi: 10.1061/(ASCE)WR.1943-5452.0000758.

[22] T. R. Sheltami, A. Bala, and E. M. Shakshuki, “Wireless sensor networks for leak detection in pipelines: a survey,” *J. Ambient Intell. Humaniz. Comput.*, vol. 7, no. 3, pp. 347–356, 2016, doi: 10.1007/s12652-016-0362-7.

[23] M. A. Adegboye, W. K. Fung, and A. Karnik, “Recent advances in pipeline monitoring and oil leakage detection technologies: Principles and approaches,” *Sensors (Switzerland)*, vol. 19, no. 11, 2019, doi: 10.3390/s19112548.

[24] “Hazen Williams formula for use in fire sprinkler systems.” https://www.canutesoft.com/Hydraulic-calculation-for-fire-protection-engineers/the-hazen-williams-formula-for-use-in-fire-sprinkler-systems.html (accessed Sep. 26, 2017).

[25] B. E. Larock, R. W. Jeppson, and G. Z. Watters, *Hydraulics of Pipeline Systems*. 1999.

[26] A. Jain and D. Zongker, “Feature Selection: Evaluation, Application, and Small Sample Performance,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 2, pp. 153–158, 1997, doi: 10.1109/34.574797.

[27] L. Yu and H. Liu, “Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution,” *Int. Conf. Mach. Learn.*, pp. 1–8, 2003, doi: citeulike-article-id:3398512.

[28] ScikitLearn, “1.13. Feature selection — scikit-learn 0.21.3 documentation.” https://scikit-learn.org/stable/modules/feature_selection.html#variance-threshold (accessed Aug. 25, 2019).

[29] H. Jiawei, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques*. 2012.

[30] N. Suguna and K. Thanushkodi, “An Improved k-Nearest Neighbor Classification Using Genetic Algorithm,” *Int. J. Comput. Sci. Issues*, vol. 7, no. 4, pp. 18–21, 2010.

[31] F. Amalina, N. Ali, N. Badrul, and A. Abdullah, “Evaluation of machine learning classifiers for mobile malware detection,” *Soft Comput.*, vol. 20, no. 1, pp. 343–357, 2014, doi: 10.1007/s00500-014-1511-6.

[32] D. Tomar and S. Agarwal, “A survey on Data Mining approaches for Healthcare,” *Bio-Science and Bio-Technology*, vol. 5, no. 5, pp. 241–266, 2013.

[33] “Big data helpen slim waterleidingnetwerk | Waterbedrijf Vitens.” https://www.vitens.com/pers-en-nieuws/nieuwsoverzicht/persberichten/big-data-helpen-slim-waterleidingnetwerk (accessed Mar. 28, 2020).

[34] S. R.P., N. V.E, and J. Amaranath, “Feasibility Analysis And Design Of Water Distribution System For Tiurnelveli Corporation Using Loop and WATER GEMS Software,” *Int. J. Appl. Bioeng.*, vol. 7, no. 1, pp. 61–70, 2013, [Online]. Available: http://www.academia.edu/5905702/Feasibility_Analysis_And_Design_Of_Water_Distribution_System_For_Tiurnelveli_Corporation_Using_Loop_and_WATER_GEMS_Software_..