A fault sensitivity analysis for anomaly detection in water distribution systems using Machine Learning algorithms

Alexandru Predescu, Mariana Mocanu, Ciprian Lupu
Faculty of Automatic Control and Computers, University POLITEHNICA of Bucharest
mircea.predescu@stud.acs.upb.ro, mariana.mocanu@cs.pub.ro, ciprian.lupu@acse.pub.ro

***Abstract*—The introduction of Machine Learning in large scale utility networks extends the room for improvement in the quality of service and maintenance costs. The ever expanding network of smart meters allows for a more accurate estimation of the state of the water distribution systems, at the same time requiring modern data processing solutions. By fusion with the more traditional approach in this field of research it is possible to enhance the existing capabilities for network analysis and to extend the algorithms to the level of cognitive abilities that form a basis for more efficient decision support system. In this paper we extend the fault sensitivity analysis for water distribution systems with the insights provided by state-of-the-art Machine Learning algorithms for data clustering and anomaly detection.**

I. Introduction

Nowadays, the improvements in the quality of service and resource management in large scale water distribution systems are in many ways related to the integration of smart infrastructure and Information and Communication Technology (ICT) systems [1].

IoT (Internet of Things) architectures and the industry standard SCADA (Supervisory Control and Data Acquisition) systems provide the background for a smart infrastructure in various fields [2].

Considering today's aging infrastructure, that requires regular maintenance and repairs, the problem of fault detection (i.e. leaks, burst pipes) is critical for ensuring minimal disruption to the service.

Using data from smart meters is valuable for reducing the effects of a breakdown, in a proactive strategy [3], and for reducing the repair time and costs by a more accurate detection of leaks and estimating the location within a complex network.

The problem of detecting anomalies in measurements structured as time-series data is of particular interest in the domain of water distribution systems. Unsupervised learning methods such as clustering are being used in different domains for extracting relevant information from vast amounts of data. The problem of anomaly detection in a given data set is a subject of both supervised and unsupervised learning. Unsupervised learning is given by clustering methods that are used to extract patterns from the data set, without prior knowledge of the possible results. The anomaly detection methods represent a type of supervised learning methods, as the anomaly can be defined as a discrepancy between the data set and the patterns given by clustering. In the context of water distribution systems, the data is recorded using smart meters and is available as time-series. In the context of ML (Machine Learning), this represents the original data set. For each meter, there are time-series grouped by days. As the data source is known, the problem of detecting anomalies is related to the particular meter. In the context of multiple meters, in a given network topology, there can be correlations between the anomalies.

The effect of a detected anomaly on the incidence of several other detected anomalies, as a direct consequence, is the main subject of this paper. Then we propose a method for large scale detection of anomalies that uses the results from different layers of the network, which is a subject of a future paper.

II. Related work

Machine Learning (ML) is currently being used in a broad range of applications to extract relevant information from large amounts of data. Data clustering and aggregation, outlier and anomaly detection are some of the applications of Machine Learning in various fields of work.

In [4] a clustering method using the k-means algorithm is used to extract patterns from different levels in a multiple clustering strategy. Using Automatic Meter Reading (AMR), the patterns for a given consumer demand can be extracted from 24-hour time-series and it is possible to create a classification of consumer types [5]. In the same way, the anomaly patterns can be extracted, using the discrepancies in the consumer demand as the input data set.

In the case of utility networks and water distribution systems, the cost of maintenance can be reduced by adopting a proactive strategy, as stated in [3]. Nonetheless, anomalies and breakdowns are bound to appear and the rapid response is a key factor in minimizing the damage and reducing the repair costs.

There are several approaches to anomaly detection using Machine Learning, such as density-based anomaly detection, clustering-based anomaly detection, support vector machine-based anomaly detection [6]. The clustering-based method we find best suited to the problem, as it can also reveal information about the types of consumers in the network and the types of anomalies that can be related to the particular consumer type.

In the context of water distribution systems, an important source of anomalies is represented by leaks. There are many approaches for leak detection, that range from dedicated hardware solutions, to real-time monitoring and automatic control systems, as described in [7], [8], [9], [10], [11].

A method for fault sensitivity analysis in water distribution system is described in [12]. This method uses a fault sensitivity matrix that shows the direct effect of leaks on measured parameters (flow, pressure). We find that the method can be used for a fault sensitivity analysis, that uses the standard deviation of cluster centroids, in the case of anomaly detection using Machine Learning algorithms.

There are, nonetheless many types of anomalies that can arise in the context of smart distributed systems such as cyberattacks and even malicious behavior of the actors involved in water distribution system management [13]. Therefore, one can identify a broad range of patterns that can be considered an anomaly, so that an extensive knowledge base has to be considered.

III. Proposed solution

The data from the smart meters is collected from a water distribution network in Milan, and is represented as time-series for each meter for each day, each having 24 samples. The data set is structured in multiple files. Each file corresponding to the associated meter, contains the data as a matrix, where the rows represent the days and the columns represent the hours (samples). Each sample represents the average flow during the last hour. In real-life conditions, the data set would be a real-time stream instead, that has to be handled by Big Data solutions, such as message brokers (e.g. Apache Kafka, Apache Spark, RabbitMQ) and real-time databases (e.g. RethinkDB). The Big Data architecture and performance analysis is not a subject of this paper though.

Also, the proposed solution is demonstrated using a limited subset of the broad range of possible anomalies, that is given by leaks. An extensive comparison of the possible anomaly sources is a subject of further research.

The data is clustered using the *k*-means unsupervised learning algorithm to obtain the general consumer patterns for the normal conditions. When an anomaly is detected, the clustering algorithm also calculates the associated patterns. By subtracting the normal conditions clusters from the anomaly clusters, it is possible to extract the anomaly patterns and to highlight the particular type of anomaly. This method assumes a form of buffering of the raw data and the clustering results, so that the results are aligned with the real-time detection and confirmation of the anomaly. The strategy for separating the normal clusters from the anomaly clusters is described in the following paragraphs.

We consider the context of a decision support system that should generate alerts when anomalies are detected in a water distribution system. The focus is on the Machine Learning algorithms that can extract this information using raw measurements from smart meters. We consider a clustering algorithm, along with the anomaly detection algorithm, as shown in Figure 1.

The main problem with this configuration is that an anomaly in the data has the effect of shaping the cluster centroids so that the anomaly is no longer detected after a certain amount of time. This is similar to a closed loop feedback system, where the anomaly can be represented by a disturbance, that is rejected as part of the normal operation of the system.

For evaluating the proposed anomaly detection method, we define the mathematical model of the network and a set of questions that should provide further understanding of the problem.

A. Mathematical model

The model of the water distribution system is a standard graph representation that implements the fundamental laws defining the water flow through a pipe. The model is used to calculate the state of the network based on the available data.

The first law is the mass conservation law that states that the input flow to a node is equal to the output flow:

$$\sum_j q_{ij} - \sum_j d_{ij} = 0,\ i = 1..n \tag{1}$$

The second law is the equivalent of Ohm's law for laminar flow and gives the value of the flow for a network segment between two adjacent nodes:

$$q_{ij} = \frac{h_i - h_j}{R_{ij}} \tag{2}$$

$$R_{ij} = \frac{8\eta l_{ij}}{\pi r_{ij}^2} \tag{3}$$

The dynamic model is simulated using a first order filter that is defined according to the physical characteristics of the associated pipe:

$$G_{ij}(s) = \frac{1}{T_{ij}s + 1} \tag{4}$$

We considered the following notations:
q_{ij} - input flow from node i to node j
d_{ij} - output flow from node i to node j
h_i - head (pressure) in node i
R_{ij} - resistance to flow in the pipe from node i and j
η - fluid viscosity
l_{ij} - length of the pipe between node i and node j
r_{ij} - radius of the pipe between node i and node j
$G_{ij}(s)$ - transfer function of the filter for the pipe between node i and node j
T_{ij} - filter time constant
s - Laplace transform variable

The transient state is simulated using the first order models associated to the network segments (3). The method of simulation represents an iterative approach that filters the data by a feedback component for each corresponding segment.

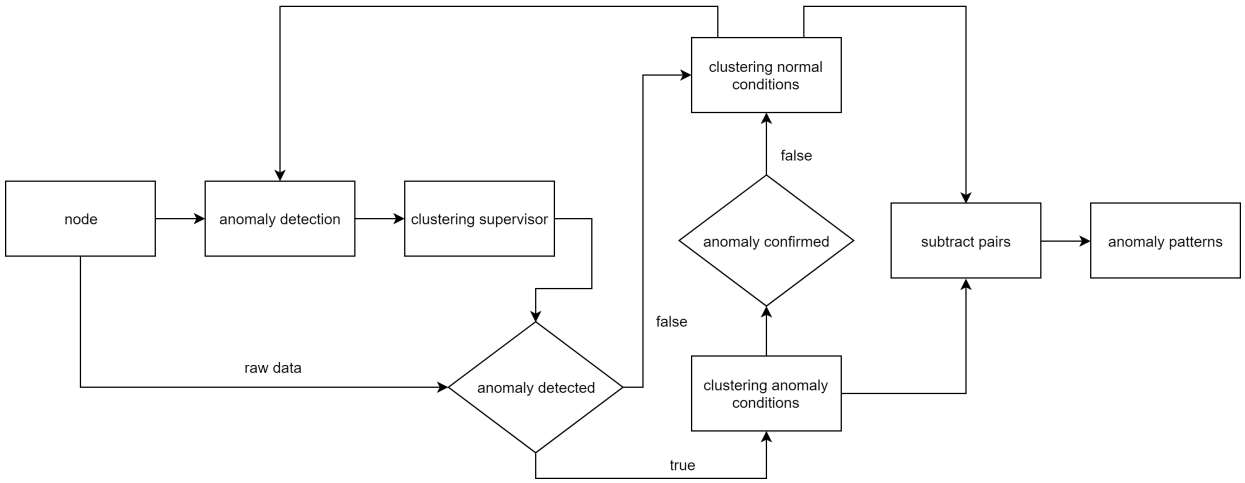

Figure 1. System overview

B. Experimental setup

We defined the configuration of the proposed anomaly detection system and the main problem that arises from the requirement of separating the anomaly from the real-time measurements.

In this paper, we answer three questions that bring an in-depth understanding of the problem and the possible solutions:

A. The first question that we analyze in section IV is the transient effect, the amount of time until the effect of the anomaly is transposed in the cluster centroids. This is an important step for the proposed method, providing a measure for tuning the parameters of the associated buffering algorithm.

B. The second question that we analyze in section IV is the correlation effect, the correlation between multiple detected anomalies. This is an important step for the proposed method, as it provides a measure for the level of uncertainty in the estimated parameters.

C. The third question that we answer in this section is related to the possibility of isolating the anomaly so that the cluster centroids for the demand patterns are not affected, and only the normal behavior is represented. This takes into consideration the results that we obtain for the first two questions in section IV.

We considered the experimental approach that is validated in simulation as follows:

1. The normal conditions are simulated using real measurements for a given consumer node and estimated data, obtained using a hydraulic model of the network, for the other nodes. The clustering results that are provided by this simulation define the reference patterns that are further used to evaluate the anomalies.

2. The anomaly scenarios are simulated by adding a specific profile to the data set that is associated to a given consumer node. The same hydraulic model is used to estimate the values for the other nodes.

3. A measure of similarity (discrepancy) is calculated from the normal conditions and the simulated anomaly scenario. There are two possibilities for this purpose:

3.A. The anomaly conditions are simulated in the same way as the normal conditions, using the data that is modified according to the type of anomaly that is evaluated. The cluster centroids, obtained separately for the normal conditions and for the simulated anomaly scenario are subtracted and the average standard deviation is calculated. This variant is actually represented in the system overview schematic (Figure 1).

3.B. The real-time data sets associated to the normal and anomaly conditions are subtracted and then we use the clustering algorithm to extract the patterns. The average standard deviation is calculated from the difference patterns. This variant we find more straight-forward to use for simulation purpose.

This setup is used for each experiment, as the first step, that represents the estimation of the anomaly profile. In this paper, we considered the method **3.B.** to measure the discrepancy between the normal conditions and the simulated anomalies. This variant has some advantages for simulation purpose. When comparing the overall clustering effort, the first variant uses two clusterings from the start. In the second variant, a single clustering is used to calculate the deviation patterns, though in the final solution, an additional clustering is still used to evaluate the patterns under normal conditions (par). When comparing the post processing effort, in the first variant, two sets of clusters have to be subtracted, after evaluating the most similar pairs. In the second variant, there is the additional estimation of the normal conditions during the anomaly (par).

It is nonetheless important that the data, that represents the normal conditions, does not contain anomalies. This is actually ensured by the DSS, in the sense that the anomaly is detected before it is transposed into the data set, and the clustering is either split into two independent processes (**3.A.**) or it is buffered until the anomaly is resolved, and the difference data

set is clustered instead (**3.B.**). This assumes that in the second variant, the normal conditions data set has to be estimated during the anomaly, by using the identified patterns (e.g. averaging). Therefore, there is no clear advantage of either variant in real-life conditions. Nonetheless, for simulation purpose, it is easier and more accurate to use the raw data, as in the second variant.

C. The correlation effect

The second question that we analyze in section IV is the correlation effect (**B.**).

The method for analyzing the effect of an anomaly on the other measurement nodes (smart meters) can be either simulated or used in a real test scenario and is represented by a fault sensitivity matrix, where the discrepancies are given by the deviation of the data set from the cluster centroids.

Considering the case of large scale networks that would benefit from a distributed approach, we propose a general method for anomaly detection across the entire network, which will be described in a future paper. So far, we define the anomaly detection algorithm for a given data source, such as AMR data from smart meters, and we calculate the correlation between the possible anomalies in the network. This approach can be generalized by a multiple-stage clustering and anomaly detection. This method is also known as a type of hierarchical clustering and is described in [14]. Considering this scenario, the anomaly detection algorithm uses the data from the current stage as the data set (cluster centroids for the current stage, raw data for the first stage) and the data from the higher stage as the reference clusters. Each stage reveals a more detailed view of the anomalies, from the top level to the individual consumer level. Moreover, this approach shows an overview of the sub-networks that are part of the water distribution system.

D. Anomaly isolation

We start by defining a general method, providing a general solution to the third question (**C.**). We propose a method to isolate the anomaly when it is detected, so that the clusters are not affected. As part of a decision support system, the anomaly is signaled to the operator, with an associated time limit for confirmation. The subsequent measurements are clustered separately in the anomaly clustering buffer.

As the anomaly can be an actual change in the system configuration (e.g. an increase in the number of consumers), the operator can confirm the anomaly or dismiss it as a normal state. In the case of a confirmed anomaly, when the cause is resolved and the system returns to the normal state, the additional clusters generated during this time frame are discarded. On the other hand, if the anomaly is dismissed as a normal state, the additional clusters are merged with the initial clusters and the clustering algorithm continues. This allows for a continuous update of the consumer demand patterns as part of a normal evolution, while not being affected by transient anomalies. Therefore, the estimation accuracy for both clustering and anomaly detection is improved by removing the cross-causality effect between the two algorithms.

These two scenarios can be visualized in Figure 1. The final decision, regarding the confirmation of the anomaly, can be the task of the operator, which can also be a higher level DSS (decision support system) that can learn from previous events. If the anomaly is confirmed, the operator delegates the task of fixing the problem and when the problem is fixed, the operator signals this to the DSS that will then dismiss the clusters generated during this time frame. If the anomaly is dismissed as normal behavior, the operator signals this to the DSS that will then aggregate the clusters.

Considering this scenario of a procedure defined for a single anomaly, it is possible to identify multiple anomalies that were in fact caused by the same problem. Therefore, it is important to evaluate the effect of an anomaly on other measurement nodes (smart meters) so that the problem can be accurately identified. This is further analyzed in section IV.

IV. Results

In this section, we answer the first two questions defined in section III by implementing the scenarios for a measurement data set provided by a real water distribution system. The implementation is done using Python and the scikit-learn library [15].

For the real-time clustering, we defined a particular implementation that updates the cluster centroids for each new data set (i.e. the measurements for the next day). There are two scenarios that are used, one with the original data set, and the second one with an additive constant deviation as the anomaly scenario.

A. The transient effect

The transient effect i.e. the amount of time until the effect of the anomaly is transposed in the cluster centroids, is the first question proposed in this paper, that aims to provide a measure for tuning the parameters of the buffering method in the anomaly detection system.

We analyze the effect of an anomaly in terms of standard deviation from the normal state, over a time frame of multiple days. Having a data set with measurements, we simulate a constant additive deviation, starting with a particular sample, and we compare the clusters obtained from the normal data set to the clusters obtained from the modified data set. The simulation model is described in III-A. We defined the dynamic model using first order filters for network sections, according to the associated physical dimensions. Therefore, the transient response is that of a first order system for the simulated flow in the corresponding network segments.

We subtract the two data sets, and the resulting characteristic is, accordingly, that of a first order system, as shown in Figure 2. The effect can be interpreted a transient response, with the settling time of less than 3 days i.e. the time until the output becomes stable.

As the sampling time of the clustering algorithm is given by the daily measurements, the system has a fast transient

response. Therefore, the clustering can be done by using a buffer, containing the clusters obtained using the latest data, that is analyzed in real time (i.e. at every sample) and is later added to the cluster centroids (e.g. every 3 days or the actual settling time for the particular scenario). The buffer is compared to the normal conditions, to check for unusual behavior, and the DSS can provide the required input, that is to add the buffer to the normal clusters or not, while the operator can override this decision, after thorough field inspection.

Therefore, we can propose, in this case, a 3 day time limit to evaluate the anomaly, before the clusters are automatically assigned to the normal conditions. If the anomaly is dismissed as normal state, the process can be finalized before the proposed deadline, and this is recommended so that the clustering in normal conditions is up to date. The time limit is proposed with the same considerations.

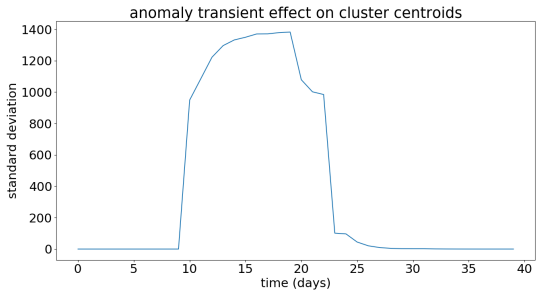

Figure 2. Transient effect

B. The correlation effect

The cross-correlation between the measurements in multiple nodes can be simulated by using a hydraulic model. In the same way, the cross-causality between the detected anomalies as a deviation between the two scenarios can be estimated.

In the first scenario we consider a demand node that is assigned the measured data and the other nodes that are assigned the estimated data from the hydraulic model. The clustering algorithm receives the data from all the nodes and generates clusters for each node, that represent the normal conditions. In the second scenario, we add the constant additional demand to the measured data and observe the effect on clusters obtained using the entire data set.

For each node, we consider the steady-state deviation (i.e. the maximum deviation within the simulation time frame) as the measure of non-similarity between the two scenarios. The method for calculating this measure that we used in this experiment consists in initial subtraction of the estimated data from the two scenarios (**3.B.**).

By using this method for simulating an anomaly in each of the measurement nodes, we obtain a fault sensitivity matrix, that shows the effect of each anomaly on the entire measurement data set as described in [12]. Therefore, this effect can be simulated and the overall sensitivity of the network can be estimated in advance. The results of this network analysis can be used to predict the most appropriate cause of an anomaly, from a range of possible causes.

In the proposed test scenario, we consider the network represented in Figure 3. The parameters are given as pressures at the supply nodes and the flow at the consumer node that is assigned the real measurement data set. The demands associated to the other consumer nodes are considered constant. The real value is not known in the simulation as it would be available in real-life scenarios through measurements. It is not necessary though, as we are interested in the actual deviation that we obtain by simulating different anomaly scenarios and not the absolute values.

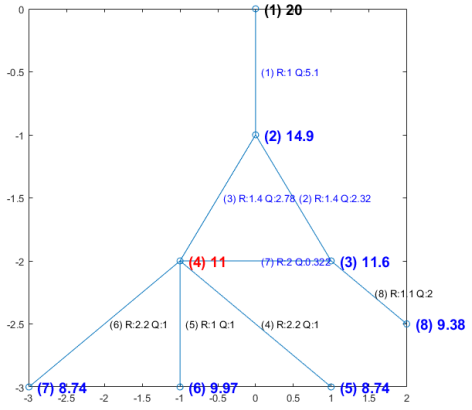

Figure 3. Network representation

Therefore, the supply pressure is constant and the demand in one of the demand nodes is given from the measurement data set. The other parameters are calculated from the model. By running this simulation over a given time frame and running the clustering algorithm on the data, we obtain the patterns for each demand node.

The standard deviation is used as a measure of discrepancy in the two scenarios. Therefore, for a given demand node there is a scalar value that is added to the fault sensitivity matrix for the particular node (column) for the simulated anomaly node (row).

By simulating the anomaly in each of the nodes and using the method described above, the fault sensitivity matrix is generated. The colormap representation shown in Figure 4 represents an overview of the network in terms of overall sensitivity to anomalies. The particular test scenarios are shown as blue dots and the higher sensitivity is shown in yellow. The sensitivity along the network segments is estimated by interpolation of the real measurement points, so that it is possible to visualize the effects in multiple locations between the actual measurement nodes.

The multiple effect of real anomalies on the measured (in this case simulated) parameters is a source of uncertainty that is highlighted by this method. Therefore, there can be multiple nodes that show a discrepancy in the measured parameters in the case of a particular anomaly.

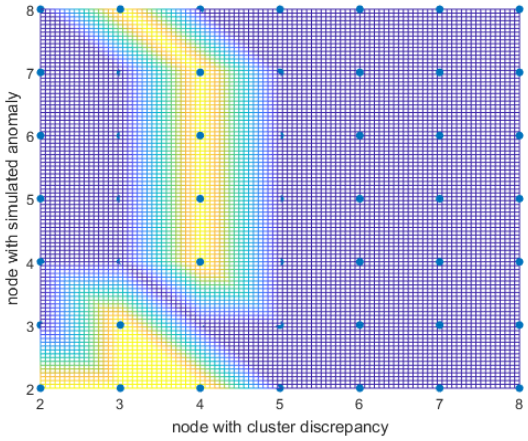

Figure 4. Network overall sensitivity

As expected, the highest sensitivity is generally obtained when the anomaly is on the same node. In this particular experiment, considering the case of a simulated anomaly in the second node, the effects are visible on nodes 3 and 4 as well. If we consider that a fault can appear along the pipe, in between the measurement nodes, then the uncertainty can be higher in finding the exact source of the problem. Therefore, a more detailed analysis, that simulates leaks along the pipes as well, can be useful.

V. Conclusion

In this paper, a method for fault sensitivity analysis in a water distribution system is combined with Machine Learning algorithms for data clustering and anomaly detection. The results validate the method for estimating the overall sensitivity to leaks using the cluster similarity measure between the normal state and the anomaly scenario. The method provides an insight for leak detection capabilities within a given network topology and allows for increasing the location accuracy in a decision support system. The effect of an anomaly on the data clustering algorithm is analyzed from a real-time perspective by measuring the transient response of the system. Starting with this result, a strategy is provided so that the clustering and anomaly detection methods can be used, both simultaneously and independently as part of a decision support system for water networks. The task of the operator for validation of provided insights on the state of the system can be delegated to a higher-level decision support system that keeps track of the breakdown history and implements a Machine Learning strategy as well. The method can be extended in the case of multiple-layer, hierarchical networks providing a separation of concerns with a corresponding amount of detail. The experimental scenarios described in this paper represent a progressive approach to the problem of fault sensitivity analysis for anomaly detection in water distribution systems.

Acknowledgement

We are thankful to the PN III Program P3 European and International Cooperation, UEFISCDI, that supported the research activity and part of the presentation in conference, as well as to the H2020 Twinning Program, that partially supported the publication under the 690900 project - Data4Water

References

[1] M. Umar and W. Uhl, “Integrative review of decentralized and local water management concepts as part of smart cities (lowasmart),” Norsk institutt for vannforskning, Tech. Rep., 2016.

[2] O. Vermesan and P. Friess, *Internet of Things - Converging Technologies for Smart Environments and Integrated Ecosystems*. River Publishers, 2015.

[3] M. Moglia, S. Burn, and S. Meddings, “Decision support system for water pipeline renewal prioritisation,” in *ITcon Vol. 11, Special issue Decision Support Systems for Infrastructure Management*, vol. 11, 2006, pp. 237–256.

[4] A. L. N. Fred and A. K. Jain, “Data clustering using evidence accumulation,” in *Object recognition supported by user interaction for service robots*, vol. 4, 2002, pp. 276–280 vol.4.

[5] D. García, D. Gonzalez, J. Quevedo, V. Puig, and J. Saludes, “Clustering and classification of aggregated smart meter data to better understand how demand patterns relate to customer type,” Universitat Politècnica de Catalunya (UPC), Tech. Rep., 2015.

[6] Oracle and DataScience.com. Introduction to anomaly detection. [Online]. Available: https://www.datascience.com/blog/python-anomaly-detection

[7] C. Lupu, D. Chirita, S. Iftimie, and R. Miclaus, “Consideration on leak/fault detection system in mass transfer networks,” in *Energy Procedia*, vol. 112, March 2017, pp. 58–66.

[8] R. Isermann, “Process fault detection based on modeling and estimation methods—a survey,” in *Automatica*, vol. 20, no. 4, July 1984, pp. 384–404.

[9] N.C. Turner (Ferranti Ltd.), “Hardware and software techniques for pipeline integrity and leak detection monitoring,” in *Society of Petroleum Engineers*, 1991.

[10] G. Geiger, “Principles of leak detection,” in *Fundamentals of leak detection. KROHNE oil and gas*, 2005.

[11] S. Oven, “Leak detection in pipelines by the use of state and parameter estimation, master thesis,” Norwegian University of Science and Technology, Department of Engineering Cybernetics, Tech. Rep., January 2014.

[12] A. Predescu, M. Mocanu, and C. Lupu, “Modeling the effects of leaks on measured parameters in a water distribution system,” in *2017 21st International Conference on Control Systems and Computer Science (CSCS)*, May 2017, pp. 585–590.

[13] P. Nader, P. Honeine, and P. Beauseroy, “Detection of cyberattacks in a water distribution system using machine learning techniques,” in *2016 Sixth International Conference on Digital Information Processing and Communications (ICDIPC)*, April 2016, pp. 25–30.

[14] A. Predescu, M. Mocanu, and C. Lupu, “A multiple-layer clustering method for real-time decision support in a water distribution system,” in *2018 21st International Conference on Business Information Systems (BIS)*, July 2018.

[15] scikit-learn developers (BSD License). scikit-learn. [Online]. Available: http://scikit-learn.org/stable/