Real-time clustering for priority evaluation in a water distribution system

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

***Abstract*—Nowadays with the development of smart infrastructure for water resource management, there is an increased need for efficient operation and management of water distribution infrastructures. In this paper, we propose a system for real-time clustering system priority evaluation in a water distribution system. Data clustering algorithms are modified for real-time scenarios providing decision support for management of water distribution system. The proposed method is compared to the standard clustering methods in terms of accuracy and real-time capabilities.**

***Index Terms*—Machine Learning, Real-Time Clustering, Water Distribution System, Decision Support System**

I. Introduction

Given the increased demand for autonomous systems as part of a smart infrastructure for water resource management, Information and Communications Technology (ICT) system gather data and aim to provide insights about the different stages of water distribution systems and to inform all the actors involved (e.g. operators, utility providers and citizens) [4].

Efficient resource management in complex water distribution systems is a subject of ongoing research. This includes the estimation of water demand from smart meter data and detection of anomalies [1]. The problem of handling the data from all these sources can be defined from multiple points of view such as storage solutions, scalable services, Machine Learning algorithms. The latter is a subject of research in the field of Artificial Intelligence, where algorithms are designed to extract relevant information from vast amounts of data such as the *k*-means algorithm which is used for data clustering and outliers detection.

The aging infrastructure in water distribution systems requires regular maintenance and an increased cost of operation. There are different theoretical and practical approaches for detection and prevention of leaks and problems in the water network such as comparing the measured data to a statistical or hydraulic model.

For water distribution systems and utility networks in general, there are different strategies for scheduling the replacement of pipes before a failure occurs which are based on the risk level (i.e. cost of the asset) and the associated probability (i.e. probability of breakdown). A proactive strategy is recommended for high risk and low probability situations while a reactive strategy is often used for high probability and low risk situations and is traditionally used as there were, until recently, limited possibilities for real-time data acquisition and processing.

978-1-5386-2205-6/18/$31.00

Considering the rapid development of Internet of Things architectures and the industry standard SCADA (Supervisory Control and Data Acquisition) systems, the quality of utility networks and services can be improved [8].

Using data from smart meters allows for increasing the efficiency of maintenance work as high priority sections can be serviced first, thus reducing the overall effect of a breakdown on the network [3].

Considering the large scale water distribution systems, a real-time decision support system should increase the efficiency of preventive measures reducing the maintenance costs and improving the quality of service.

Clustering means grouping similar data into homogeneous groups, without having knowledge of how groups are defined [14]. The main criteria of grouping objects is similarity. So, the objects that have maximum similarity with other object are grouped together and objects with minimum similarity are grouped in other groups. Clustering method is used mainly for exploratory data analysis, as it identifies structures and generates summary of data. Also, clustering can be used as pre-processing stage for other data processing tasks.

In this paper we study the possibility of combining time-series clustering method with a real-time component as part of a decision support system.

The structure of the paper is as follows: In Section I is presented a short introduction, is Section II are presented the relevant works related to the subject, in Section III is presented our proposed solution about clustering for priority evaluation in a water distribution network, in Section IV are presented the experimental results and finally in Section V are presented the conclusions of the paper and future work.

II. Related work

The ever expanding water distribution systems has become the subject of ongoing research as there are different problems given by the size, complexity and aging infrastructure. The main focus is the improvement of the quality of service to the consumer while reducing the operational cost. As the aging infrastructure has lead to an increase in maintenance cost, many research papers focus on the leak detection possibilities that use the available data from smart meters as an alternative to more costly, hardware based solutions such as those described in [9]. Leaks are detected when there are discrepancies between calculated and measured values, an approach similar in the case of mass transfer systems [6]. However, this approach requires an accurate model of the hydraulic system and there is a requirement of high accuracy for leak detection. This implies a higher cost for sensors and measurement systems [7].

Time-series clustering methods have been applied in many domains such as aviation/astronomy as a pre-processing step for outliers detection in astronomical data [15], biology by functional clustering of time-series [16], climate for discovery of climate indices [17], energy for discovering energy consumption pattern [18], finance for finding seasonality patterns [19], medicine for detecting brain activity [20], psychology for the analysis of human behavior [21], robotics by forming prototypical representations of the robot experiences [22] and user analysis for analyzing emotional behavior of users in social networks [23].

In [10], the *k*-means algorithm is used to accumulate evidence from a given data set using a multiple clustering strategy. The demand patterns of different consumer types can be extracted from AMR (Automatic Meter Reading) meters which provide 24-hour time-series as described in [2]. This can be extended in the case of multiple layers in the network providing increasing levels of detail.

In [13] the authors propose a server-side approach using an online algorithm for clustering geodata for online maps. The geodata clustering is based on location in order to improve visual analysis and to improve situational awareness. The proposed approach works in real-time and could be used for clustering of massive geodata for online maps in reasonable time.

III. Proposed solution

In this section we present an overview of the solution as part of a decision support system. The details of implementation and validation of results through extensive testing and comparing with a standard method are described in section IV. We propose the integration of a clustering algorithm into a real-time decision support system for prioritizing pipe replacements in a water distribution system. The pipes will have an associated priority that is calculated using the real-time data from the measurement nodes. As there is a high probability that there are nodes that are not equipped with meters, the missing data has to be inferred in one of the stages of the algorithm. The real-time requirements assume the collection of data at 1-hour intervals from all the measurement nodes. We consider three test scenarios to validate the proposed solution:

A. The control case, where all the data is known in advance and a standard clustering algorithm is used.

B. The 1-day sampling case, where the clusters are updated with daily time-series. In the first experiment, we consider the test scenario using the standard algorithm for the first half of the data set, and 1-day sampling for the second half.

C. The 1-hour sampling case, where the clusters are updated with 1-hour samples. We define this as partial recalculation of centroids. In the first experiment, We consider the test scenario using the standard algorithm for the first half of the data set, and 1-hour sampling for the second half. This is considered the proposed method for real-time clustering that we validate in this paper.

We compare the results that we obtained using the last two scenarios to the results obtained using the first case in terms of the standard deviation. This shows the accuracy of the proposed method for long-term measurements.

We consider that all consumer nodes and none of the intermediary nodes are equipped with meters. As the data from the intermediary nodes is not available there are two possibilities. The first would be to estimate the data using a hydraulic model and then to run the clustering and priority evaluation algorithm for each node. The second is to estimate the priority of the nodes by using the calculated priorities of the consumer nodes in a bottom-up approach.

The second method is proposed in this paper providing an abstraction over the otherwise difficult modeling problem. Moreover, the processing requirements are lowered as we are not interested in the real-time estimated raw data. The decision support system is based on two subsystems which handle the real-time data clustering and the priority estimation (Figure 1).

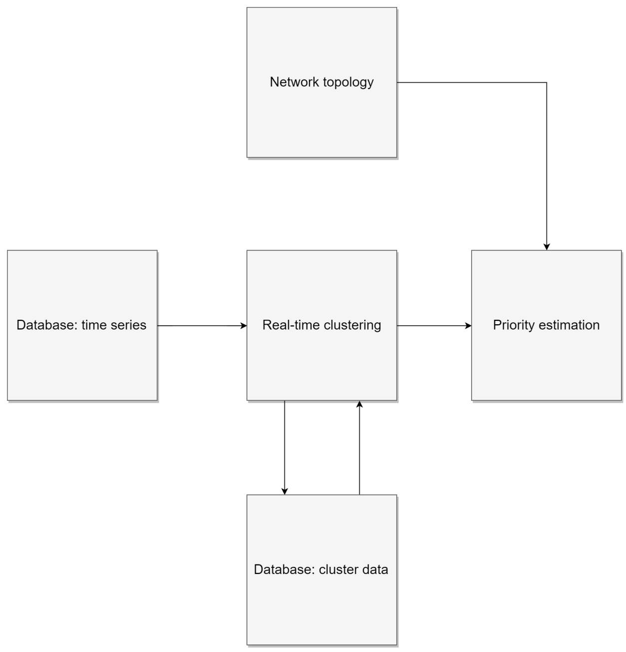

Figure 1. System architecture

The *Real-time clustering* subsystem handles the data clustering and the assignment of real-time data samples from the database to a cluster. As the recalculation of clusters from most recent time-series is done at regular intervals or when the estimation requirements are no longer satisfied, the current cluster data is stored into the database as well. Real-time data and the corresponding node is then assigned to one of the identified clusters.

The *Network topology* subsystem uses the network topology and the assignments of consumer nodes to clusters and calculates the priority of each network segment. This produces the results that will be directly used in the decision support system.

The proposed solution uses the data from consumer nodes as an array of time-series (for each day). We assume that the consumers are either residential, industrial or mixed and we propose a method to calculate the priorities from partial time-series in order to assess the network state in real-time.

The network is represented as a directed graph, having supply nodes, consumer nodes and intermediary nodes. The consumer nodes provide the measurements and the associated priorities can be calculated as a function of the demand profile and the consumer type. The associated priority for a pipe is calculated as the average priority of the adjacent nodes, where the priority of the intermediary nodes is calculated as the average priority of their child nodes.

The data is structured into 24-hour time-series which represent the daily consumer demand. There are two stages in the proposed solution. The clusters are updated at a predefined interval (i.e. number of time-series) and the time-series are classified in real-time for assignment to a cluster and possible detection of anomalies. The real-time component can be defined either as complete or partial time-series assignment. In both cases, the algorithm finds the matching cluster by using a similarity measure (e.g. euclidean distance).

In order to satisfy the real-time application requirements, the Database subsystem is designed to support concurrent of reading and writing with low latency, scalability and availability, efficient storage and access methods. We chose to use a time-series Database which is optimized for time-stamped or time-series data. Time-series represent measurements that are tracked and aggregated over time. In our case these are sensor data.

Cassandra [12] is an open-source distributed database system, that supports rich data structure and powerful query language. It has the following characteristics: schema flexible, support range queries, high scalability (e.g. a single point of failure does not affect the whole cluster). Cassandra offers different methods for data ingestion such as, **COPY FROM** command for ingesting CSV data, **sstableloader** for loading large data and **BulkOutputFormat** for streaming Hadoop data.

IV. Results

We used Python and the *scikit-learn* package for the application server. The clustering is done using the k-means algorithm: `kmeans=KMeans(n_clusters=3)`. In the case of new time-series which have to be added to a cluster after the initial clustering, an additional parameter is used for the k-Means algorithm: `init=centroids`. This initializes the algorithm with the previous results and ensures the same order when updating the clusters. It is however not possible to initialize with single samples and to do a partial update of the centroids.

For network representation, the *networkx* package is used. The topology is defined by using data from a GIS (geographic information system) server. We used an AngularJS application with data that is retrieved from the server using HTTP requests and shown according to the requirements of a decision support system for monitoring the state of a water distribution system in real-time.

The nodes and edges are shown in Figure 2 using a color-map representation according to the calculated priorities. The scale is also proportional to the node priority.

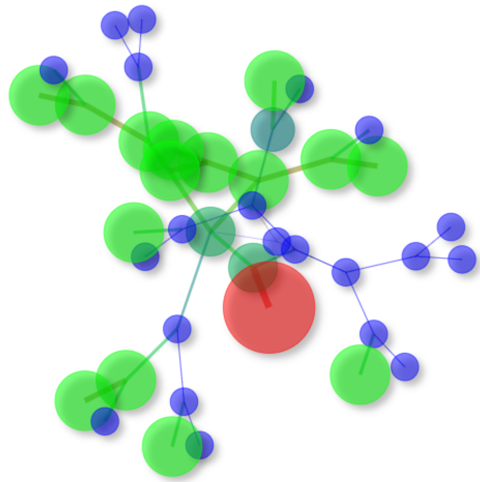

Figure 2. Network overview

The entire data set is provided by 21 measurement nodes where the data for each node consists of 82 days of measurements, each day being represented by a 24-hour time-series with a sampling time of 1 hour.

A. Real-time clustering

Considering the 1-hour sampling test case (C), we find that the default clustering algorithm from the *scikit-learn* library does not allow partial recalculation of centroids. Therefore we created a Python implementation of the k-means algorithm as follows. The algorithm is initialized with the current clusters obtained from previous measurements. The time-series obtained from new measurements (i.e. 1-hour samples) are compared with the cluster centroids (at each sample) in terms of similarity. Considering that the sampling is done at regular intervals, the euclidean distance should suffice. Otherwise, another method for comparing the time-series should be used such as DTW (dynamic time warping). The nearest cluster is found and the corresponding centroid is partially updated. First, we run the standard clustering algorithm for the first half of the measurements and then we use the remaining measurements as 1-hour samples with our proposed algorithm. The centroids obtained this way are then compared to the control case (A).

The results are shown in Figure 3. The first chart shows the results for the control case (A). The second chart shows the results for the 1-day sampling scenario (B). The third chart shows the results for the 1-hour sampling scenario (C). In each case we considered two clusters.

The deviation from the control case (A) is shown for both test scenarios (B and C) to provide an accurate comparison. The 1-hour sampling scenario (C) shows a deviation of about 50% higher than the 1-day sampling scenario (B).

Considering that the shape of the resulting centroids and the standard deviation does not show a significant loss of information, the method provides adequate results. We further validate this assumption by extensive testing using multiple data sets from different locations (consumer nodes). When comparing the results obtained using multiple data sets from 21 locations and the same method we obtain the average deviation shown in Figure 4. The deviation from the control test (A) for the 1-day sampling scenario (B) is shown in orange and for the 1-hour sampling scenario (C) is shown in blue. This shows consistent results in terms of the relative deviation when running this experiment several times, with the 1-hour sampling scenario (C) showing an average deviation of about 40-50% higher than the 1-day sampling scenario (B).

So far, we considered the three test scenarios with the first half of the data set used for the initial clustering. Further, we compare the results obtained using the same method, with a different number of time-series for the initial clustering. We consider the test cases as the entire range between 2 and 81 time-series (i.e. number of days used for the initial clustering). This range is defined considering the use of 2 clusters which require at least 2 samples in the data set.

The results are shown in the surface plot in Figure 6 for the 1-day sampling scenario (B) and in Figure 7 for the 1-hour sampling scenario (C). In both figures, the horizontal axis represents the nodes and the vertical axis represents the test case (having an increasing number of time-series used for initial clustering, from 2 to 81). The deviation from the control case (A) is represented on the z-axis which can be seen as a color-map representation of the data. As expected, the deviation is higher with a lower number of time-series used for the initial clustering in both cases and it is higher in the 1-hour sampling scenario (C) when compared to the 1-day sampling scenario (B).

This suggests that there is a trade-off between the accuracy of the clustering algorithm and the real-time requirements. Therefore, there is a limit to the amount of time (e.g. number of days) that the algorithm can run before the degradation of the accuracy. In this case, a full recalculation of clusters becomes necessary at regular intervals.

B. Priority evaluation

We propose the method described above for real-time priority evaluation with a sampling time of 1 hour. Therefore, We validate the 1-hour sampling method (C) by loading 1-hour samples in sequence from a 24-hour time-series corresponding to a consumer node. The samples are then compared to the current clusters using a similarity measure and then associated with the best match.

The results (Figure 5) show that it is difficult to use the actual measurements for an accurate estimation of the consumer priority because of the uncertainties in the consumer demand which can indicate a different cluster at different times of day. For this reason we suggest that the individual consumer patterns should be updated in real-time and then compared to the consumer patterns for the entire network. This implies a partial recalculation of centroids for the individual consumers at predefined intervals.

Considering this result, instead of the current measurements which do not provide stable results for the cluster assignment, we use the clusters updated in real-time using the partial update algorithm (C). This provides generally higher accuracy for real-time assignments.

V. Conclusion

In this paper, machine learning algorithms are used in the context of water distribution systems as the main part of a real-time decision support system. The proposed method combines a two-stage clustering algorithm with a partial update of the individual consumer patterns obtained in the first stage. This allows for an improved response time, providing information about the consumer demand pattern at regular intervals during the day. As part of a proactive resource management strategy, a fast response time for priority evaluation represents an improved efficiency of resource spending by addressing the most important and the most critical assets first and within a reduced amount of time.

The clustering algorithm is designed to allow 1-hour sampling time, as compared to the standard method that requires the entire time-series for updating the centroids. The actual calculation of centroids is done using a partial recalculation with the real-time data as a 1-hour sample from the 1-day (24 sample) time-series. The results that were obtained using this method and a measurement data set show that the method can be used effectively for the defined real-time constraints and there are multiple possibilities of improving the accuracy for the particular scenario. The effect of the number of samples used for the initial clustering is examined through an extensive test scenario that reveals the relative accuracy to the standard clustering method. Also, the deviation is consistent for the entire measurement nodes, suggesting the overall performance of the proposed solution.

The method can be used for each individual measurement node as part of a complex water distribution system, providing insights on the behavior of the network components. As the decision support system requires an overview of the system as a whole, this data can be further used in a second-stage clustering that reveals information about the entire network. The solution can also benefit from parallel computing as the dynamic updating of existing clusters for consumer nodes can be handled by a decentralized architecture and the local results can be aggregated at the upper levels.

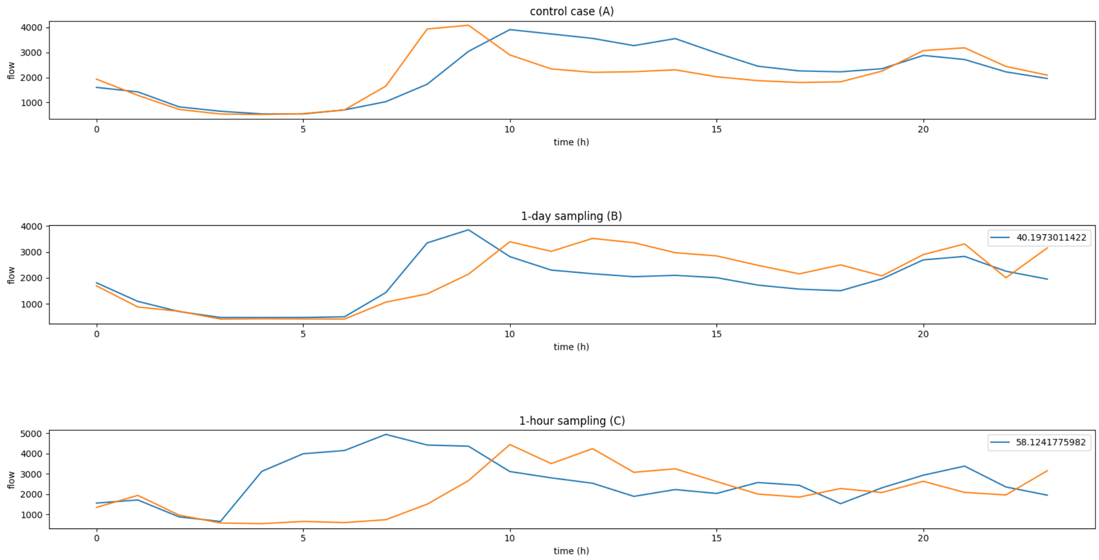

Figure 3. Clustering method comparison

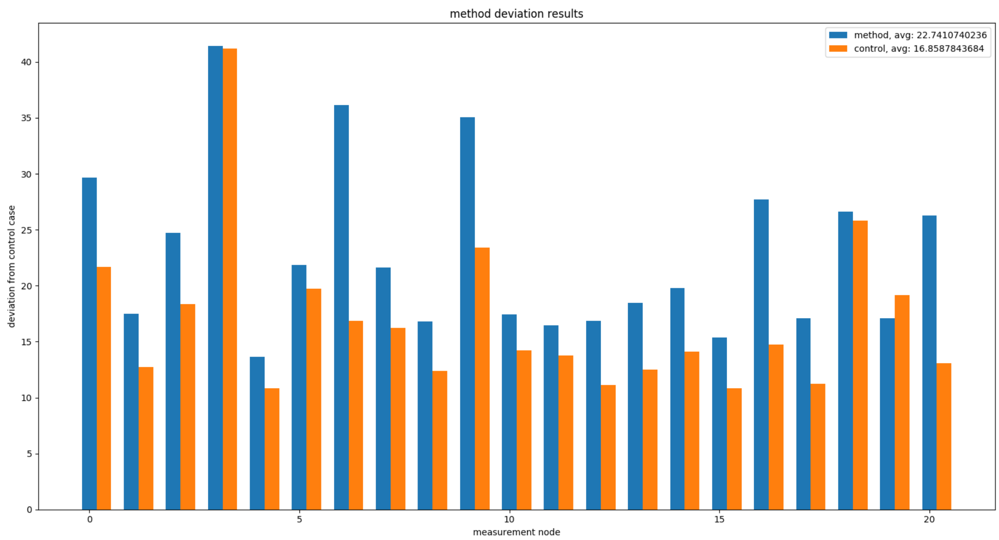

Figure 4. Deviation for multiple locations

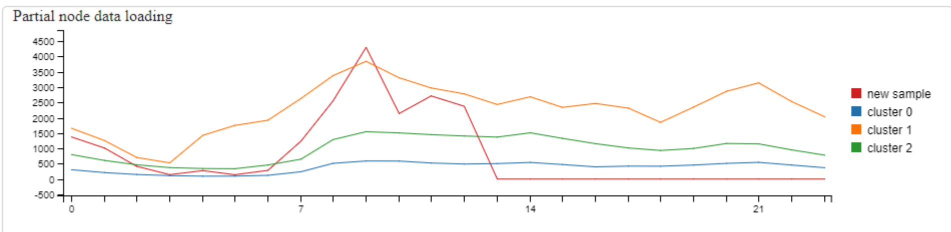

Figure 5. Cluster assignment using 1-hour samples

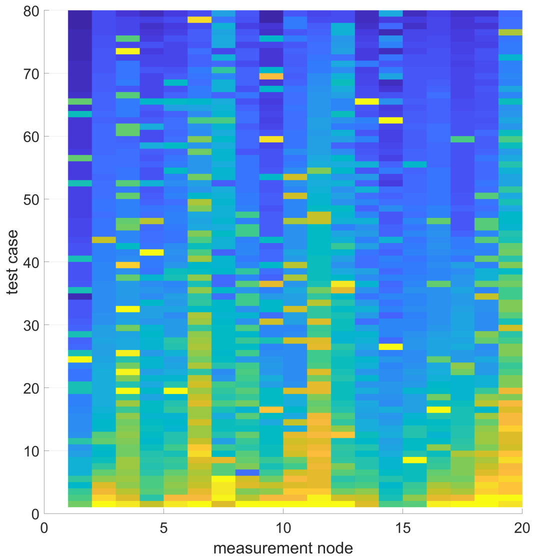

Figure 6. Deviation with 1-day sampling

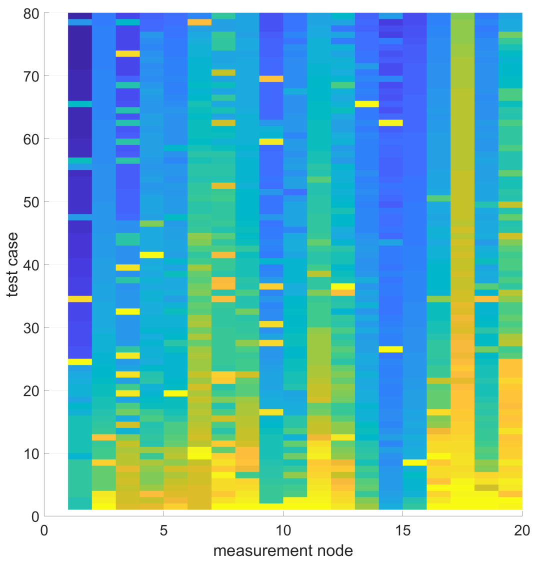

Figure 7. Deviation with 1-hour sampling

ACKNOWLEDGMENT

Research was supported by UEFISCDI, through the PN III project no. 16/2016, Awarding Participation in H2020 - Data4Water, no. 690900

REFERENCES

[1] D. Garcia, D. Gonzalez, J. Quevedo, V. Puig, J. Saludes *"Water demand estimation and outlier detection from smart meter data using classification and Big Data methods"*, Conference report 2015.

[2] D. García, D. Gonzalez, J. Quevedo, V. Puig, J. Saludes, Universitat Politècnica de Catalunya (UPC) *"Clustering and classification of aggregated smart meter data to better understand how demand patterns relate to customer type"*, Conference report 2015

[3] Magnus Moglia, Stewart Burn, Stephen Meddings, *"Decision support system for water pipeline renewal prioritisation"*, 2006, ITcon Vol. 11, pg. 237-256

[4] Umar, Muhammad and Uhl, Wolfgang *Integrative Review of Decentralized and Local Water Management Concepts as Part of Smart Cities (LoWaSmart)*, 2016, Norsk institutt for vannforskning

[5] Felix Iglesias * and Wolfgang Kastner, *"Analysis of Similarity Measures in Times Series Clustering for the Discovery of Building Energy Patterns"*, 2013, Energies Vol. 6, pg. 579-297

[6] Ciprian Lupu, Doina Chirita, Serban Iftimie, Roxana Miclaus, *"Consideration on leak/fault detection system in mass transfer networks"*, Sustainable Solutions for Energy and Environment, EENVIRO 2016, 26-28 October 2016, Bucharest, Romania

[7] Alexandru Predescu, Mariana Mocanu, Ciprian Lupu, *"Modeling the effects of leaks on water distribution systems"*, 2017, CSCS21, Bucharest, Romania

[8] Ovidiu Vermesan, Peter Friess, *"Internet of Things - Converging Technologies for Smart Environments and Integrated Ecosystems"*, River Publishers

[9] N.C. Turner, *"Hardware and Software Techniques for Pipeline Integrity and Leak Detection Monitoring "*, 1991, Society of Petroleum Engineers, Offshore Europe, 3-6 September, Aberdeen, United Kingdom

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

[11] M. A. Alsheikh, S. Lin, D. Niyato and H. P. Tan, *"Machine Learning in Wireless Sensor Networks: Algorithms, Strategies, and Applications"*, IEEE Communications Surveys & Tutorials, vol. 16, no. 4, pp. 1996-2018, Fourthquarter 2014.

[12] Lakshman A, Malik P., *"Cassandra: structured storage system on a p2p network"*, In Proceedings of the 28th ACM symposium on Principles of distributed computing 2009 Aug 10 (pp. 5-5). ACM.

[13] Amirkhanyan, A., Cheng, F., & Meinel, C. (2015, November). *"Real-time clustering of massive geodata for online maps to improve visual analysis"* In Innovations in Information Technology (IIT), 2015 11th International Conference on (pp. 308-313). IEEE.

[14] P. Rai, S. Singh, *A survey of clustering techniques*, Int. J. Comput. Appl.7 (12) (2010)

[15] U.Rebbapragada,P.Protopapas,C.E.Brodley,C.Alcock,Finding anomalous periodictimeseries,Mach.Learn.74(3)(2009) 281–313.

[16] N. Subhani,L.Rueda,A.Ngom,C.J.Burden, *Multiple gene expression profile alignment for micro array time-series data clustering*,Bioinformatics 26(18)(2010)2281–2288.

[17] M. Steinbach,P.N.Tan,V.Kumar,S.Klooster,and C.Potter, *Discovery of climate indices using clustering*,in:Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery And data Mining, 2003,pp.446–455.

[18] F.Iglesias,W.Kastner, *Analysis of similarity measures in times series clustering for the discovery of building energy patterns*,Energies 6 (2) (2013)579–597.

[19] M.Kumar,N.R.Patel, *Clustering seasonality patterns in the presence of errors*, in:Proceedings of Eighth ACMSIGKDD,2002,pp.557–563.

[20] A. Wismüller,O.Lange,D.R.Dersch,G.L. Leinsinger,K.Hahn,B. Pütz, D. Auer, *Cluster analysis of biomedical image time-series*,Int.J. Comput. Vis46(2)(2002)103–128.

[21] V.Kurbalija, J.Nachtwei, C.Von Bernstorff, C.von Bernstorff,H.-D. Burkhard, M.Ivanović, L.Fodor, *Time-series mining in a psychological domain*,in:Proceedings of the Fifth Balkan Conference in Informatics, 2012,pp.58–63.

[22] M.Ramoni, P.Sebastiani, P.Cohen, *Multivariate clustering by dynamics*, in:Proceedings of the national Conference on Artificial Intelligence,2000,pp.633–638.

[23] J. Zhu, B.Wang, B.Wu, *Social network users clustering based on multivariate timeseries of emotional behavior*, J.China Univ.Posts Telecommun 21(2)(2014)21–31.