WATER LEAKAGE DETECTION AND LOCALIZATION ANOMALY MATRIX: A DETERMINISTIC APPROACH

Subhrajeit Bhowmick[1], Klaus Seifert[2]
[1] *subhrajeit.bhowmick@cubalytics.de,* [2] *klaus.seifert@cubalytics.de*
[1] Cubalytics GmbH Im Sonnenrech 16, 65366 Geisenheim, Germany

ABSTRACT

In an endeavor to early detection and localization of pipe leakages based on sensor readings installed across a drinking water distribution network, we took a deterministic approach using Python to design a novel method that we termed as the *"Anomaly Matrix"*.

The network comprises of 905 distinct pipes/links with 119 sensors installed across it, which detect pressure, flow, level and demand. Thus, not all pipes in the network have sensors installed at either end. This led to an assumption that a leakage will reflect incongruity at least at the nearest connected sensors at either end of a leaking pipe.

Note:

All naming conventions and values in the Diagrams are for illustration purposes only.

T1, T2, T3… Tn indicate Timestamps.
P1, P2, P3… Pn indicate Pipe/Link IDs.
C1, C2, C3… Cn indicate Column names indicating Pressure/Demand/Flow/Level Sensors Nodes.

As a first step, an algorithm was developed in Python to identify the closest sensors at either end of each pipe within the network. Thus, for each pipe, the algorithm propagates through n pipes of one end and m pipes of another end, for a pipe with two openings.

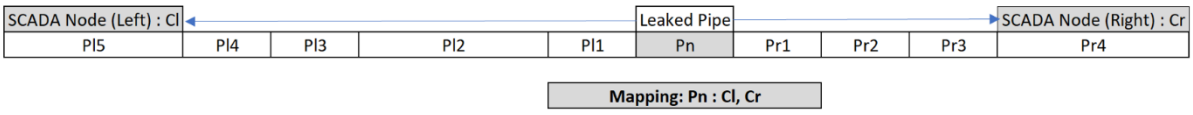

Figure 1. Illustration: Algorithm to identify nearest located sensor nodes to a leaked pipe

Next, an ***in-memory Master Map*** is created where all pipes and their corresponding nearest sensor nodes are mapped.

Nodes of Pipes Master List	
Pipe ID	**SCADA Node ID**
P1	C1
P2	C1, C2
P3	C2, C4
…	…
…	…
…	…
P30	C45, C56
P31	C23, C24
P32	C66, C79, C11, C25, C45
P33	**C3, C8**
P34	
…	…
…	…
P66	**C1, C7, C8**
P67	**C9, C10**
P68	**C1, C8, C10**
…	…
P75	C34, 44

Figure 2. Illustration: In-Memory Master Map associating each pipe with its nearest sensor

Demand, pressure, flow and level data for the year 2018 were combined together based on Timestamp and scaled to prepare a master data set for analysis.

T	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
T1	1.5	4	3	7.5	3	1.5	3	3	3	3	7.5	3	1.5	3	7.5	3	1.5	3	3	3
T2	0	1	4	5	3	2	3	0	0	0	0	0	0	0	0	0	0	0	0	0
T3	0	7.5	3	1.5	7.5	1	7.5	7	1.5	1	2	4.6	7.9	1.5	3	6	3	6	0	0
T4	0	7.5	0	0	0	0	0	0	5	1	6.8	7.5	4	3	2	3	8.3	9	0	0
T5	7.9	3	3	4	5	8	9	3	8	2	6.8	1	1.5	7.5	1	1	1	0	1.5	0
T6	7.9	2	1.5	7.5	3	1.5	6.8	4	3	3	6.8	1	1	4	5	0	2	4	6.8	8
T7	7.9	1.5	6	2	5	2	2	2	6	6.8	1	7.9	7.5	3	1.5	7.5	1	3	6.8	8
T8	7.9	2	9	0	7	9	6	6	6	6.8	1.5	1	1.5	7.5	3	5.5	5	2	6.8	8
T9	0	0	0	6.8	3.6	5.6	9	0	0	6.8	4	1	4	1	6	2	6	1	0	0
T10	0	0	4	6	0	0	0	1.5	0	3	6.8	2	3.7	1	9	7	9	0	1.5	0
T11	0	3.5	3	9	4	0	7	6	6	1.5	6.8	2	1.5	7.5	0	9	0	0	6.8	1.5
T12	7.5	1	2	0	3	2	3	1.5	7.5	3	6.8	6	6	6	4	4	4	2	6.8	6
T13	0	3.5	1	4	5	5	2	1	4	5	8	2	4	2	2	2	3	0	6.8	6
T14	7.5	3	7.5	3	1.5	7.5	1	7.5	3	1.5	7.5	1	1.5	4	2	1.5	2	2	2	6
T15	0	3.5	0	2	0	0	0	0	7.5	3	4	1	4	5	5	2	1	0	1.5	6
T16	7.5	8	0	1	6	5	4	1.5	7.5	1.5	7.5	7.5	3	1.5	7.5	1	3.6	5.6	2	7.9
T17	0	1	4	5	3	2	1	4	5	3	2	1	1	4	5	2	2	1.5	3	6
T18	1.5	7.5	3	1.5	7.5	1	7.5	3	1.5	7.5	1	1	7.5	3	1.5	7.5	1	3	4	6

Figure 3. Illustration: Master Data Set

Using Python, the data set was transformed into a binary matrix where 0's and 1's represented normal and anomalous values respectively from the sensors in the network. These anomalous values are the statistical outliers. i.e., values beyond 1.5 times the Inter-Quartile Range.

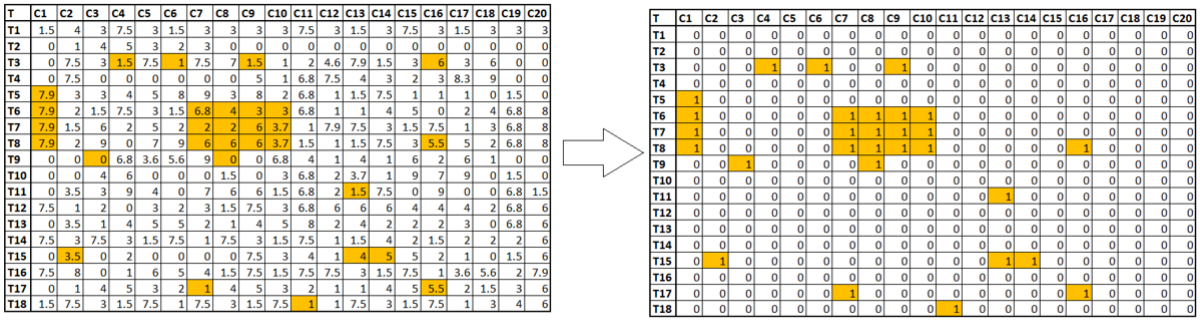

Figure 4. Illustration: Identification of Anomalies and Transformation into Binary Matrix

The dimensions of the binary matrix were then reduced by discarding rows that contained only normal readings – resulting in a binary matrix with one or many anomalous records in each row. We termed this matrix as "Anomaly Matrix", which was the basis of detection and localization of leakage instances.

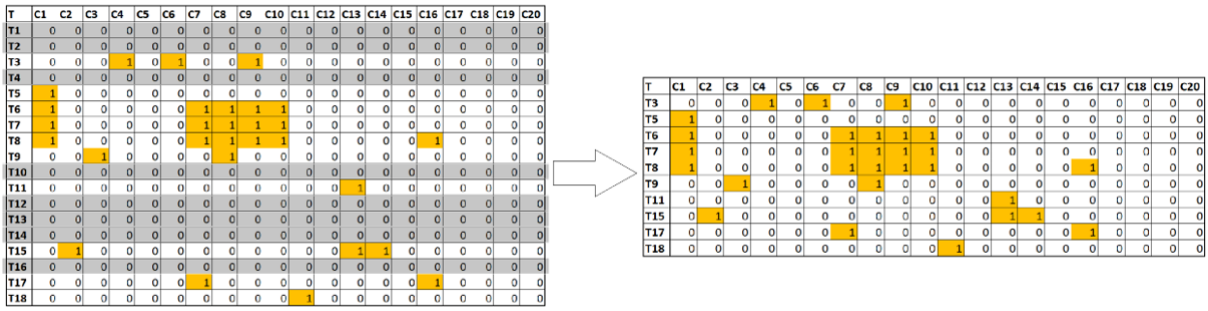

Figure 5. Illustration: Removing 0-value rows to build the Anomaly Matrix

A Checksum was introduced to compute the sum of each row. This Checksum would signify the number of simultaneous anomaly occurrences in one or multiple sensor nodes at Tnth instance.

T	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	Checksum
T3	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	3
T5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
T6	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5
T7	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5
T8	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	6
T9	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2
T11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
T15	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	3
T17	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	2
T18	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1

Figure 6. Illustration: Introduction of Checksum

Anomalous data points that occurred at regular/continuous time intervals were identified. This interval was calibrated by analyzing the patterns from the known leaked pipes from the 2018 Leak Report. In the following illustration (Figure 7), Timestamp T9 is also considered because on T9th instance, there were multiple (two) anomalous readings, and it also is in succession of previously recorded anomaly timestamps T6, T7 and T8.

T	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	Checksum
T3	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	3
T5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
T6	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5
T7	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5
T8	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	6
T9	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2
T11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
T15	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	3
T17	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	2
T18	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1

Figure 7. Illustration: Identification of frequent time intervals in Anomaly Matrix

Another algorithm was introduced to identify the associated sensor nodes that detected irregularity at a particular timestamp and then recognize clusters of such repeated instances. The frequency of repetition was calibrated based on the observations from the known leaked pipes shared in 2018 Leak Report.

T	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	Checksum	List of Nodes
T3	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	3	
T5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
T6	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5	C1, C7, C8, C9, C10
T7	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5	C1, C7, C8, C9, C10
T8	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	6	C1, C7, C8, C9, C10, C16
T9	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2	C3, C8
T11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	
T15	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	3	
T17	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	2	
T18	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	

Figure 8. Illustration: Associating respective List of Nodes against each Checksum

All possible combinations of the identified sensor nodes were established for each instance. These combinations were cross-validated with the in-memory Master Map (Figure 2), and only the valid combinations were considered for each instance of Timestamp in the Anomaly Matrix.

This resulted in further elimination of rows that did not formulate to a valid sensor node combination.

T	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	Checksum	List of Nodes	Valid Node Combinations
T3	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	3		
T5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1		
T6	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5	C1, C7, C8, C9, C10	(C1, C7, C8), (C9, C10), (C1, C8, C10)
T7	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5	C1, C7, C8, C9, C10	(C1, C7, C8), (C9, C10), (C1, C8, C10)
T8	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	6	C1, C7, C8, C9, C10, C16	(C1, C7, C8), (C9, C10), (C1, C8, C10)
T9	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2	C3, C8	C3, C8
T11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1		
T15	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	3		
T17	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	2		
T18	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1		

Figure 9. Illustration: Transforming List of Nodes into Valid Node Combinations

Respective group of Pipe identifiers were then stored against each row as Pipe-Clusters (Figure 10 – Column, “Pipes”) by cross-referencing the valid sensor node combinations with the Master Map (Fig. 2).

																					Leakage Detection			Leakage Localization
T	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	Checksum	List of Nodes	Valid Node Combinations	Pipes
T3	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	3			
T5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1			
T6	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5	C1, C7, C8, C9, C10	(C1, C7, C8), (C9, C10), (C1, C8, C10)	P66, P67, P68
T7	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	5	C1, C7, C8, C9, C10	(C1, C7, C8), (C9, C10), (C1, C8, C10)	P66, P67, P68
T8	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	6	C1, C7, C8, C9, C10, C16	(C1, C7, C8), (C9, C10), (C1, C8, C10)	P66, P67, P68
T9	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2	C3, C8	C3, C8	P33
T11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1			
T15	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	3			
T17	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	2			
T18	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1			

Figure 10. Leakage Detection & Localization

Finally, for every leakage instance, an algorithm determined the leaked pipe as the one with maximum difference between Actual Demand and Base Demand in each Pipe-Cluster, after execution of a Pressure-Dependent Demand simulation where Emitters were introduced to simulate leakage.

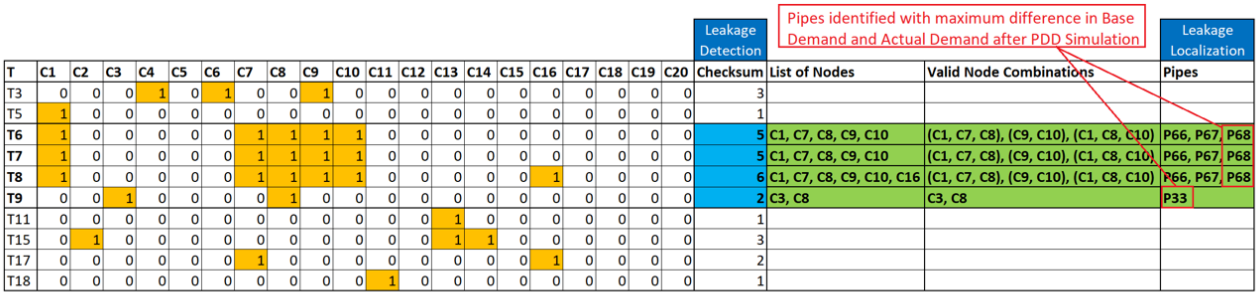

Figure 11. Final Step for identifying Leaked Pipes

While calibrating, algorithm detected the following start times for the reported 2018 leakage instances.

	linkID	endTime	firstAnomaly	timeToFix
0	p461	2018-04-02	2018-03-15	18 days
1	p232	2018-02-10	2018-05-16	-95 days
2	p673	2018-03-23	2018-05-16	-54 days
3	p628	2018-05-29	2018-05-17	12 days
4	p538	2018-06-02	2018-03-05	89 days
5	p866	2018-06-12	2018-05-17	26 days
6	p31	2018-08-12	2018-05-29	75 days
7	p183	2018-09-01	2018-04-03	151 days
8	p158	2018-10-23	2018-05-17	159 days
9	p369	2018-11-08	2018-05-18	174 days

Figure 11. 2018 Leakage Detection

We assume that the couple of instances in blue (Figure 11) had leakage origination in 2017, and thus the first detection in 2018 was at a later date to the end time.

Keywords:
Anomaly Matrix, Checksum, In-Memory Master Map, Pressure Dependent Demand (PDD), Python

SUMMARY

The leakage patterns were not identical in their statistical properties in 2018 and 2019, with the later year experiencing nearly continuous fluctuations. Hence a deterministic approach was chosen over a probabilistic approach in solving the problem.

As each pipe has two openings, any leakage would lead to anomalous measurement in the closest located sensors on either end of the pipe. Hence timestamps that simultaneously recorded anomalies in multiple sensors were considered in designing the Anomaly Matrix, while the ones recording anomaly in a single sensor were discarded.

Any irregularity in a pipe P will have a cascaded effect across all the pipes located between the closest sensors at each end of the pipe P. Therefore, these sensors cannot directly pin-point a specific leaked pipe, but identify a cluster of pipes located between these sensors. Thus, considering each set of these pipes as a Pipe Cluster, the pipe with maximum difference between Base Demand and Actual Demand is identified as the leaking pipe after performing a Pressure Dependent Demand (PDD) simulation.

Our results indicate that, amongst the 12 leakage incidents identified in 2019, most of these originated due to occurrence of one or more major incidents within the network near mid of May 2019. Also, almost all the leakages were initiated between 07:30 and 08:00 hours in the morning, indicating a regular instability during this period causing the leakages.

We will continue application of this concept further by developing each element in the Anomaly Matrix with transformed data points by using core principles of Hydraulics Engineering. Also, outlier fencing, which was currently considered as 1.5 times the Inter-Quartile Range, can be further calibrated if we have information about sensor qualities like accuracy and precision.

Thus, this deterministic approach using the concept of *Anomaly Matrix* can be used as a robust methodology in early detection and localization of water leakages across different water network ecosystems.