CreaTe Graduation Project: Strategic Placement of Water Analyzers

A.A.A.Mahran

July 2024

I Dedicate this to my family, friends, and teachers, as without their endless love, Incredible support and golden patience, I would simply not be in the place I'm in today.
Thank you all

Contents

1 Introduction

As far as humans have existed, water management has been an essential part, as humans need clean water for survival. It can even be said that some of the great civilizations of the past have been characterized by their water system, for example, think about the Romans aquaducts or the irrigation system of the ancient Egyptians, etc. A great deal of effort has gone into designing, building, and maintaining these systems. The creators of such systems share an obligation to design the systems in such a way that the population can enjoy water, with as little hesitation about its quality. While throughout history, humans have gotten better at providing the population with increasingly higher quality water, it remains a goal to strive for better. This is also reflected in the UN Sustainable Development Goal 6.[45] Water purification and monitoring still remain important to this day. more recently, an emerging topic has become the analysis and attempted removal of Disinfection By-Products (DBPs) in the Water Distribution System (WDS).

DBPs are a result of chlorine reacting with natural organic matter, which often presents itself in the form of biofilm inside the pipes of a water distribution system. The growth of DBPs inside a network is not understood well, Which is why sensors are used to predict the occurrence of them inside a network. DBPs are harmful to health because they are carcinogenic.[53] Therefore, it is important to supply the residents of cities with chlorine-based water distribution purification systems with clean, potable DBP-free water. The measuring of DBPs can be done using sensors[54], [55] spread out through a network. It is important to note here that these sensors only will provide data for a single node in a network. Placement, operation, and maintenance of these sensors is costly. We cannot place the sensors everywhere throughout a network and therefore we should consider the location of these sensors carefully. This means that their placement should be optimized to reduce the costs of the system, while maintaining its monitoring quality as much as possible.

The underlying goal of this graduation project is to provide cleaner water to residents of cities with networks that use chlorine as their main water purification method. To be able to reach this goal, this project will focus on the optimisation of sensor placement to detect the occurrence of DBPs in a water distribution system.

1.1 H2OforAll Project

"H2OforAll project aims to assess main Disinfection By products (DBPs) sources through the development of fast, cost-effective and accurate sensor monitoring devices and by modelling their spread through drinking water distribution systems. DBPs toxicity and environmental impact will be studied and measures will be proposed to protect drinking water chain."[28]."Breakthrough water treatments to remove DBPs or avoid their formation during water disinfection processes will be developed, paying attention to their life cycle analysis, costs and risks. A Central Knowledge Base with reliable data on the occurrence of DBPs in the EU and their effects will be created to increase awareness and engagement of society and governmental organizations about these drinking water contaminants and favour new policy responses and guidance."[48]

The H2OforAll project is a Europe-wide project. With 18 Partners, including Universidad de Coimbra [48], WETSUS [29] CYENS Centre of Excellence [15] being involved. [28]

The H2OforAll project will run a trial test in the city of Coimbra, which includes a trial run with water monitoring sensors to monitor the water distribution. This trial run will place sensors around a water distribution system of a city that currently resides 1200 to 4000 residents [47].A goal of this trial run is the Measuring of the disinfection by-products that may be present in the waters of their system. While the sensor locations and types are most likely already set in stone and /or not all available for sensor placement due to restrictions outside of water distribution modelling for the case pf Coimbra, This Graduation Project will take the trial test of Sensor Placement around the water distribution network of Coimbra as a template of how sensor placement is generally done, but will consider the case where the sensors can be placed anywhere over the network. This assumption is made, so that the goal of finding an optimal solution over a network is not preemptively constrained, without having shown its true potential. Being unlinked from the constraints of this specific case, also will ensure that a possible solution found for this case will have a more far-reaching impact, since it now will also be more applicable to other water distribution systems.

2 Background Research

The literature review of this topic has two functions and so it will be presented with these two parts in mind. The first part of this literature review serves to be an introduction to relevant topics surrounding WDN modelling, gaining the reader an understanding of subjects relating to this graduation project. The second part of the literature review discusses related work similar to the problem at hand. It serves to gain a deeper understanding of how this problem has been approached in the past, what limitations such approaches have seen, as well as what parts of this research have been left unaddressed. Despite these two parts of the literature have been presented in two separate sections,this knowledge was gained in a rather cyclical way, learning of the existence of a concept, followed by understanding that concept, going back and forth between writing and learning

2.1 Background Information

2.1.1 WDN

According to [19] a water Distribution network is the collective system that delivers water consisting of pipes, reservoirs, pumps, to deliver water to the population these include homes, hospitals, schools, and industries. As these water distribution systems are actively developed, changes in pipes (size, length, materials) as well as their methods of construction are made. These developments, along with regular wear of the established system can reduce the water's quality, by way of breaches, and/or intrusion of compounds into a water distribution network.

To understand the behaviour of a water distribution network, The Water distribution network is abstracted in the form of a graph, where the nodes represent junctions between pipes, and the links represent the pipes themselves. in this abstraction, information about the pipes, such as their size, length, topological height, and inner roughness are incorporated in the links. The nodes of a water distribution system, especially the endpoints, carry the information on how much water is demanded through that certain node. This "nodal demand" parameter is calibrated to how much water each endpoint of a water distribution network consumes, or how much they might consume given a specific scenario.[14] mentions "Based on this and previous studies, the writers believe that not only tanks but dead ends in distribution systems are very significant in terms of both water-quality and contaminant-propagation modeling."

2.1.2 Chlorine disinfection

The Centers for Disease Control and Prevention (or CDC for short) defines chlorination as ”the process of adding chlorine to water to kill parasites, bacteria, and viruses”[49]the paper also mentions that chlorine levels up to 4 ppm are safe for drinking water[49].The CDC also mention periodically switching to chloramine being an already implemented solution to remove biofilm from a water distribution system, as well as chloramine lasting longer in the water distribution network and generating less disinfection by-products. [14] mentions that ”a simple first-order decay model associated with modeling chlorine residuals is inadequate”, signaling that water distribution systems are complex systems that need careful modeling in order to be accurate to real-life scenarios. According to the WHO, ”Chlorine is the most widely used primary disinfectant and is also often used to provide residual disinfection in the distribution system.” [51]”Residual concentrations of chlorine of above about 0.6 mg/L or more may cause problems of acceptability for some consumers on the basis of taste, depending on local circumstances” [51]”Chlorine reacts with naturally occurring organic matter in raw water to form a range of unwanted by-products.” In chapter 8—2 of [51]by the WHO according to[40] There are three major types of reactions that lead to the consumption of chlorine in the bulk of water: oxidation, addition, and substitution. The paper mentions that addition and substitution reactions are sources of chlorinated DBPs, but Oxidation reactions are not.

2.2 Disinfection-By-Products

The decrease in biofilm cell count is partly attributed to low water temperatures in the months of April and June, as shown in Fig. 3 (b). The results suggest a higher biofilm growth rate in uPVC pipes in the GCWDS at high temperatures ($> 20°C$) during summer in contrast to the lower temperatures in winter. [40] [30] experimentally measures DBP (THMs,HAAs, and HaNs) formation, and its relation to **pH**. it mentions that HaNs, despite their low concentrations are the most toxic, followed HAAs, with THMs being the least toxic. The article states that ”at pH - 6.0 had the highest cyto- and genotoxicity, which decreased with increasing pH. ” the paper suggests that the pH does not go below 7, since the formation of more toxic DBPs increases. Table 1 of [54] charts a safe concentration limit for classes of DBPs in drinking water, according to legislation in countries/continents. Here it can be seen that the EU does not have strong regulation laws for DBPs, as they only have established thresholds for 2/11 compounds listed.

2.3 Sensors in A WDN

[54] discusses traditional methods for the detection of DBPs. It mentions Cyclic Voltammetry (CV) to be a Common method of signal analysis, along with Square wave Voltammetry (SWV) The EPA lists water quality param-

eters that can be measured with water quality sensors: Temperature sensor, pH, dissolved oxygen,conductivity, Oxygen Reduction Potential, Flow Sensor, NOMs (using spectrometry) are mentioned among many other parameters [7]. In table 2.2 of [19] The EPA lists what instumentation is used for water quality parameters. In table 5.1, changes for each parameter that are considered significant are also mentioned. this list will be consulted when assessing the cost of a sensor configuration. in chapter 6 of[52] mentions water safety parameters that they deem important for measuring and modelling. they list disinfectant residual, flow, pH, pressure, temperature, turbidity, chemical and biological parameters as the parameters of focus. this chapter shall be taken into consideration when considering the parameters to model. according to the EPA "Worldwide, many agencies have historically limited the range of pH values of distributed water between 6.5 and 8.5" even though that is not enforced by law [michael shock,EPA]

2.4 Sensor Placement Optimisation steps

Many take modelling the relevant network into EPANET As a first step. this is often followed by analysis of leak detection or chlorine detection, by placing sensors around the network some literature bases itself around optimization of this multi-reagent detection problem. an extension to this problem is the modelling of interactions between reagents in the water, which further adds computational complexity to this problem.

2.5 WDN modelling

- the following section summarizes usual modelling tools relevant to this project.
EPANET(-MSX) [20] [24] the standard, widely used in research programs for Water distribution modeling. In this program, you can create a network that is analogous to your real-life-WDN. After filling in the characteristics of the system, a simulation can be run, to see how a WDN behaves.
Plenty of literature considered the case of leak loss or detection of a contamination event. the goal of these simulations is to find the optimal location, such as to measure a network with a predetermined handful of sensors. EPANET has an extension, which is made for the simulation of multiple species. This tool is useful for modeling multiple compounds which travel through the water and interact with each other.
TEVA-SPOT Threat ensemble Vulnerability Sensor Placement Optimization Tool [25], is a tool that makes use of EPANET-MSX capabilities, and adds the functionality of Placement and optimisation of Sensors through a network. this tool will be used in the assessment of the placement of the sensors in the network.

2.5.1 Chama

[23] Chama is an sensor placement package that is developed by Sandia labs and the US EPA. Chama has advanced capabilities to modify the sensor detection thresholds for multiple sensors as well as place sensors along a network, uses a mix of sensor technologies "allowing the user to use a specified mix of sensor technologies." [36] On top of this, Chama provides tools to assess sensor placement based on budget constraints. while it is based on EPANET and shares many similarities in functionality, it is made with intention of being a more general tool. it includes multiple sensor technologies, which include mobile and stationary sensors, point detectors, and Cameras. Chama does not limit itself to optimization of Water networks, as its software can also be used to detect gas leaks or seismic events, as the documentation suggests[35]. It is not obvious whether the software has the capabilities to simulate multiple compounds and the interaction between them.

2.5.2 WNTR

(Water Network Tool for Resilience)[26] it is, like EPANET (-MSX) and Chama also developed by Sandia Labs and The US EPA. WNTR is a python package, that is compatible with EPANET.it is designed to test the resilience of water networks, by including capabilities to include disruptive events, such as pipe breaks, earthquakes, power outages, fires, etc. As well as their probability of damaging the water system, and even response and repair strategies towards these calamities. This software leans more towards vulnerability assessment of Water distribution networks, but is also able to simulate water quality parameters. It is not clear whether the software has the capability to simulate multiple compounds and the interactions between them.

2.5.3 MATLAB S-PLACE toolkit

[44] is a sensor placement toolkit that uses MATLAB to compute where to place sensors to detect contamination events. the S-PLACE toolkit is built by the Eliades and Polycarpou from the KIOS Centre of Excellence in Cyprus, is more "user friendly" (By way of providing a GUI instead of a programming interface). The toolkit has the capability to model single as well as multiple species and interactions thereof since it compatible with both EPANET and EPANET-MSX. On top of this, it is programmed in MATLAB, which is a well-known, well-respected software across Engineering and Scientific disciplines. The MATLAB S-Place allows for optimisation using different functions, which allows for optimisation using different algorithms, which allows the user to assess their optimisation strategies.

2.6 Single vs Multi objective Optimisation

throughout the literature there are two problems that are most present: these are Single objective and Multi-objective optimisation problems.

Single objective problems focus on specifically one parameter (hence the name), a solution to this problem means that a sensor network has been optimised for this parameter and this parameter only. in the literature, these problems often are leak detection, chlorine detection, or water quality detection. In the form of a single parameter to define quality. often the single parameter to optimize for is TD, VC, DL or DC

Multi-objective approaches, on the other hand, seek to find a more global answer for an optimisation problem. instead of a single "best case" answer, it finds multiple solutions to a situation and finds a "Pareto front" a set of solutions for which one of the parameters cannot be optimised further, without decreasing at least one of the other parameters. This front represents the 'best tradeoffs' between multiple parameters, making sure to not unnecessarily sacrifice other metrics' performance in order to have a gain in the said parameter. From this Pareto front the best cases can be selected, based on already developed "Multi Criteria Decision Analysis (MCDA) [4] methods such as **ELECTRE TRI**[27], **K-Means clustering**[33] and **PROMETHEE**[8]. these problems are equivalent to the placement of sensors to maximize network coverage, Time to detect, and Volume consumed at the same time.

2.7 Multi-Criteria decision analysis (MCDA)

"Multi-Criteria decision analysis (MCDA) comprises various classes of methods, techniques and tools, with different degrees of complexity, that explicitly consider multiple objectives and criteria (or attributes) in decision-making problems." [4] MCDA tools are ubiquitously and used throughout many different fields of study, including, water engineering, Healthcare, Marketing, energy systems, and many more. MCDA tools are a viable means of increasing the effectiveness of decision-making, either by grouping similarly performing points or by way of outranking the different options against each other during the literature review of related work, the following MCDA tools have been encountered.

2.7.1 K-Means Clustering

K-Means Clustering[33] is a method to cluster data based on their proximity to each other, it is among the most famous MCDA algorithms. [13] K -means clustering partitions the data in (user-defined)K clusters by picking random points and calculating the distance of each data point to this these K randomly selected points. This process is repeated multiple times, after which the clustering with the most equal variance is picked. The clusters represent data points that are closely linked together, meaning they most likely have similarities in their features and performance. This in the end results in the data set being divided into groups, which each have a separate "property".

2.7.2 Outranking methods

Outranking methods compare pairs of alternatives and assign them a degree of preference, dominance, or indifference, based on how well they satisfy the criteria and their relative importance. Some of the most common outranking methods are **ELECTRE, PROMETHEE, and TOPSIS**.[42]

2.7.3 ELECTRE TRI

ELECTRE TRI (ÉLimination Et Choix Traduisant la REalité) translated.("Elimination and Choice Translating Reality")[27] ELECTRE is a family of MCDA tools, that has been applied throughout many different research fields, but mainly natural resources and environmental management, as can be seen in fig 2 of [27] "ELECTRE TRI, TRI-C and TRI-nC are for the *sorting problematic*, also called *problematic* β, in which the objective is to assign alternatives to a set of pre-defined categories. " [27]. there is software (packages) available that makes implementation of this software easy such as [17]

2.7.4 PROMETHEE

PROMETHEE(Preference Ranking Organisation Method for Enrichment of Evaluations [8] [9] PROMETHEE is a set of MCDA tools that help to find a good balance between alternatives, by assigning a weight and a score to properties, ranking them by pairwise comparison based on those and assigning a phi+ and phi- (phi -minus) score to each alternative, which identifies how much a choice is preferred and disliked, respectively. There exists software that speeds up this process of calculation and pairwise comparison.

2.8 Optimization objectives

Multiple studies consider the case of a contamination event in a water distribution network, a case where spontaneously a toxic agent is present in the water. here, researchers try to minimise the damage in multiple ways:
[5] is a review conducted in 2018 of the state-of-the-art of contaminant warning systems. it describes commonly used parameters in Optimal sensor Placement (OSP) in WDNs rather well; the following paragraphs are small summaries of the objectives that are relevant for sensor placement, as described in the literature review.

2.8.1 Time to detection (TD)

Time to detection - or TD is seen as a very important parameter in Sensor placement, as it represents the time from generation/injection of a compound to detection by a sensor of a compound. As this can be different for different scenarios, the average detection time of multiple scenarios throughout the same network is a feasible idea.

2.8.2 Demand coverage (DC)

Demand coverage is denoted as the nodal demand that is surveilled by a sensor or sensor set. The parameter explains how much of the endpoint nodes are being surveilled by the sensors, and how much of the demand is being covered. Important to note is that it only considers the nodes with a non-zero demand, meaning those that actively ”tap water” from the water distribution system. A high demand coverage means that more sensors are placed near the leaf nodes in a network, while a lower demand coverage indicates that the sensors are closer to the ”center” of a network, meaning where their sources are. the paper also indicates that DC is likely correlated with TD. (bigger DC means longer TD)

2.8.3 Population exposed (PE)

Population exposed(PE) - describes the population that is exposed to a compound, prior to detection. The PE is calculated on the basis of the nodal demand, sums the demand by all the leaf nodes that the compound has gone through, and translates that into how many people have been affected on the basis of that.

2.8.4 Extent of contamination (EC)

The extent of contamination (EC) is defined as the total length of all the pipes that are contaminated. This is used in [32], with the reason being that in case of detection, it is unclear to what extent the contamination is within the WDN, hence the assumption that the entire pipe is contaminated with the compound is made.

2.8.5 Volume Consumed (VC)

Volume Consumed is taken as how much of the **contaminated** water has been consumed by the population prior to detection. it is a straightforward concept that makes sense as a realistic parameter to minimize.

2.8.6 Detection likelihood (DL)

Detection likelihood (DL) - according to the source, is the proportion of events where a contamination is detected, it mentions that this is the definition as of A. Ostfeld and E. Salomons have been recurring names throughout the literature review.

2.8.7 Unified normalized objective

(UNO) the paper also explains an objective described as the Unified normalized objective which is the ”combination of the four (TD, PE, VC, and DL.) objectives into a single objective.” with a reference to a study performed in 2008. Others took other parameters into this optimisation question. Researchers considered the following parameters

2.8.8 Distance between sensors

Distance between sensors[6] the case describes that the distance between sensors has been considered since it is relevant to the transmission energy needed by the sensors.
Next to most used parameters [5] continues to mention also the nature of the problems presented, as well as several case studies that have been performed, which will be taken into consideration, along with a self-performed literature review.

2.9 Algorithms

Prior to the installation of Sensor Networks over a water distribution system, The implementation of various configurations needs to be tested, hence the need for modeling, of these water distribution networks. As the size of a network increases, meaning with each extra node added, so does its complexity, in this case meaning that the amount of permutations for possible placement configurations increases exponentially. Because of this increase, a brute-force search for optimal placement quickly becomes infeasible, signaling the need for a different approach to sensor placement. This sentiment has been explicitly mentioned by [6]. Several Algorithms, which seek to explore this vast search space in a more effective manner have been developed and can also be applied to the problem of sensor placement optimizations. The following section will explain shortly the workings of each algorithm.

2.9.1 Genetic Algorithms

First, an introduction to the class of algorithms that each of these algorithms belongs to: Genetic Algorithms. A genetic Algorithm, According to the MATLAB documentation, ”The genetic algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm selects individuals from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population ”evolves” toward an optimal solution.” [39]. Genetic algorithms are often used to efficiently traverse large search spaces, to come to a near-optimal solution.

2.9.2 NSGA-II

Non-Dominated Sorting Genetic Algorithm[16] or the NSGA, is a well-known, powerful Genetic Algorithm for problems that are non-dominated (also known as Pareto-optimal). The NSGA-II is a genetic algorithm that starts off with a ”population” of random data points that are ”mated and mutated” (pairwise recombined with a random element) so that the population doubles in size. Each member of the population is then evaluated in performance w.r.t.

its target indicators, where the worst scoring performances are discarded until the population shrinks back to its original size. The remaining members of this population get to mate and mutate again, in the new cycle of this process. This repeats for a preset amount of ”generations”. Each generation iterates towards a better solution, ensuring that the final answer, if implemented correctly, converges on a good, near-optimal answer.

2.9.3 GRASP

[21] Greedy Random Adaptive Search Procedure, or GRASP, is, in essence, a Greedy algorithm. That is, it will always look for the closest highest-scoring solution at each iteration. The downside to this is that for the same starting points, the algorithm will always provide the same answer. The GRASP algorithm counters this, by introducing a random element to the Greedy heuristic. In GRASP, the algorithm will consider a handful of well-scoring points, and then randomly pick between them. This adds variability to the answers, which is able to give better and worse answers compared to greedy, and avoids always converging on the same answer from the same starting point.

2.9.4 PSO

(Particle Swarm Optimisation) [34] is an algorithm that uses multiple points in a search space in a similar way as a ”particle swarm”. Each point in the algorithm keeps track of its own best answer, as well as the global best that has been found so far. The algorithm will add these parameters to each other, and based on those a new position for each data point be chosen. By gradually decreasing the size of the steps taken, a near-optimal solution will be found, provided that the algorithm has been implemented correctly.
As of now, it is assumed that all of these algorithms are capable of finding a solution that is close to optimal. After testing the performance of these algorithms. they will be ranked for their performance.

2.10 Summary of Background information

The beginning background research cover the topics of What A WDN is, the disinfection of WDNs by chlorine, the Generation of DBPs, The monitoring of WDNs using sensors, their parameters and significant deviations of those values, What tools are being used in WDN modelling, as well as the strenghts that each program has. We learnt that optimisation problems in WDN modelling often is either single- or multi- objective, and if it is the latter, that MCDA tools can help cluster the solutions or reduce the amount of choices by outranking methods. Several Optimization objectives that are prevalent in the literature are presented, the same goes for several algorithms

2.11 Related Work:

2.11.1 THE BWSN

[41] The Battle of the Water Sensor Networks (BWSN) is a design challenge, conducted by A. Ostfeld, J. Berry, E. Salomons, among many others. The paper compares the effectiveness of 15 Optimal sensor placement algorithms, all developed by separate research for resilience against a deliberate injection of a contaminant, such as to simulate a terrorist attack on a water distribution network. It does this by optimizing for 4 objectives, those being TD, PE,VC ,DL, over 2 networks, BWSN1 and BWSN2, over which it will place 5 and 20 sensors, respectively. The Paper uses EPANET 2.0 to evaluate the algorithm's performances for the different objectives, along with custom software, to speed up the process of evaluation of the different algorithms.[E.Salomons website] The paper summarizes the approaches that have been taken for this problem and recognizes that this is inherently a multi-objective problem, although it recommends that TD, PE, and VC to be taken as one objective, against DL and to present it as a 2- objective optimisation problem, because the prior (TD, PE, VC) are positively correlated with each other, and all negatively correlated with DL. This paper has presented multiple algorithms, objectives, and example networks. As well as a good template for testing the performance of these networks, which has since been reflected and reused in literature many times. Since the paper also lists a myriad of attempts at this water network sensor placement optimization problem, these will be taken into consideration as well.

[37] proposes real-time quality of the water by way of training Gated-graph Neural Network to predict the water quality at all nodes across a WDN and uses edge betweenness centrality as a method to place the sensors. The paper employs EPANET as a simulation tool to generate the training and testing data. It also uses this as an application to the city of Yantian in China.

[6] uses Information Quality, Communication Capacity, and Sustainability as objectives for its optimisation of sensor placement. The study does this with the goal of providing sensor data over a vast distance, with application to the Ergene river in Turkey. It uses the GENS framework to place sensors and mentions a multi-stage optimisation method: "first, an optimal solution for a specific objective is found. Then, the solution is updated in subsequent iterations to improve it also for other objectives."

[11] uses EPANET and Matlab to optimise sensor placement for leak detection. It uses **PSO** and **GA algorithms** to place a handful of sensors (1-5) in the **Hanoi** and **Limassol** example networks. It mentions that PSO works faster than GA, but that GA finds better solutions, meaning they have "higher efficiency" as the complexity of the networks increases.

[38] Uses the **Hanoi**, **Fossolo**, **Net3** and **Modena** networks as example networks to estimate its state (pressure and flow throughout networks) it uses **WNTR** to generate the simulation data and selects the most informative node in each cluster by way of an information entropy calculation. Which it places a sensor at afterwards.

[56] uses a **Greedy Heuristic algorithm**, as well as a Branch and bound algorithm to find the optimal sensor placement on the **BWSN1** and **BWSN2** Networks. It uses **EPANET** and **EPANET-MSX** to calculate contaminant concentration in a network over time. It assumed all nodes in a network as equally likely for an injection. using **average consumption (VC)**, as their optimization parameter and mentions that the Greedy Heuristic is effective at finding 5 & 20 sensor locations for minimizing this parameter. claiming it to have found the global optimum in a single iteration. It mentions the BWSN being a Benchmark to have measured against.

[18] mentions to use a **centrality-guided**, multi-objective optimisation technique. They mention the use of **NSGA-II** as their multi-objective optimisation algorithm and integrates the centrality into its mutation operation, meaning that the mutation operator that generally happens in the NSGA is not truly random, but biased in favor of centrality nodes. The study optimizes for the two objectives of **TD** an **Detection Network Coverage**. (DNC) this study uses **EPANET** software, to study the **BWSN1** example network. it claims that with sensors on 31% - 35% any contamination in the network can be detected. i.e **DL** = 100%, but also mentions that this percentage and amount of sensors are very high for a network, especially as they grow to bigger sizes.

[10] Optimises for water quality, using a multi-objective and Multi-criteria approach, as their title suggests. This paper uses **EPANET-MSX** and **EPANET** to model the reaction between Parathion and free chlorine inside the water bulk. they mention their MOGA being the **NSGA-II** algorithm and mention the use of **ELECTRE TRI** to cluster the Pareto optimal solution into ordered classes. this paper uses the objectives **# of sensors**, **DT** and **DL** (although it is called Probability detection) on the D-Town Network. it uses the ELECTRE Method to make 4 clusters from the Pareto front of 173 solutions. It mentions the implementation of reverse analyses, as well as adding uncertainty to the demand of the "demand nodes" as a method for improvement

[32] to identifies contaminant intrustion location in the k1 example network using data from the **BWSN1 Competition**. it uses the **PROMETHEE** MCDA method to rank the solutions. this study "proposes a sensor placement optimization method that considers different contamination probabilities. The

results showed that different contamination probabilities of the various contamination events resulted in different optimal sensor placement schemes."

[12] uses **EPANET** to find leaks in the **Hanoi** and **Limassol** Example Network with the objectives being leak detectibility, isolability, and localisation. The paper confirms that an exhaustive search is infeasible, by doing a semi-exhaustive search of the placement of 2 sensors throughout the Hanoi network(considered to be a small network, having small complexity) and assessing it as demanding. It recognizes that GA algorithms require a post-treatment analysis, although implicitly referring to MOGA. the paper further mentions that traditional optimisation methods only optimize for the current state of the model. After this, they implement their own algorithm which by - in their words "Integrating a time horizon, a more informative distance-based scoring and the possible variations of leak magnitude, our method provides a solution configuration for sensor placement with a higher level of confidence." interesting in this paper is that inclusion and varying levels of noise changed sensor placement.

[43] uses **EPANET** and **WNTR** to implement a Multi-objective Optimization problem for sensor placement of 4 sensors on the **Hanoi** network. considering the **TD** and its standard deviation Stddev(TD) as optimisation parameters. the choice of Stddev(TD) as optimisation parameter is interesting, as instead of minimal time, it makes sure that the cases have a close (TD), which most likely also has an effect on the StdDev of the VC. This paper uses the Wasserstein Distance, which it claims performs better than NSGA-II in terms of hypervolume and coverage, especially for lower generation counts.

[22] considers a method for leak detection, by way of state-space modeling, which is a , claims does not need further information and thus modeling, other than the the network and its (physical) characteristics, and the pressure and flow throughout it. in the paper EPANET is used, but only for its base use of computing effort and flows. the networks in this paper are net1 and Hanoi

2.12 Summary Of Related works

In the related works we learn the general approach to WDN modelling, which includes what algorithms, Objectives, MCDAs, networks and # of sensors are used. We gain a general approach of how WDN modelling is performed in the literature. We learn that an important moment in this Fields lay in the BWSN design challenge, where multiple research groups presented their work to compete over a same network, furthermore, we implicitly learn that these papers consider sensors in a heavily abstracted way, that is more akin to a switch being turned on.

Table 1: Summary of Related works

Author	# of Sensors	Algorithms	Tools	Objectives
A.Ostfeld et al.	5 & 20	multiple	EPANET	TD, VC,PE,
Mankad et al.	-	PSO	WNTR, EPANET	Entropy
Z. Li et al.	20-30	Betweenness Centrality,	EPANET	
Diao et al.	1-45	CG-NSGA-II	EPANET	TD, DL
Z. Hu et al.	-	PROMETHEE method	EPANET	
Y. Zhao et al.	5 & 20	Greedy, BBA	EPANET(-MSX)	VC
MV. Casillas et al.	3	GA	EPANET, MATLAB	localisation,
CVC Geelen et al.	-	Observability gramian	EPANET, WNTR	centrality
SM Cardoso et al	-	NSGA-II	EPANET-MSX,	
B Brentan et al.	-	ELECTRE TRI, NSGA-II	EPANET-MSX, MATLAB	# of sensors,
MV. Casillas et al.	1-5	(GA), (PSO)	MATLAB, EPANET	
A Ponti et al	-	MOEA/WST	WNTR, EPANET2.0	TD, StdDev(
S Ahmad et al.	30	QR decomposition	-	Energy, dista

3 Ideation

3.1 Generation of Idea:

Since the project of modelling disinfection-by-products in a network is already well-constrained, and the fact that there is a real-life case study has been no use of a specific methodology or structured framework to generate ideas.
The predefined goal of the Graduation Project being the ”Strategic placement of Water analyzers”, as well as that same goal being defined in the relating H2OforAll project (WP 2 of [28]) already defined a great deal of what my work will be within the project: to design a tool or present a methodology to aid in the decision making process of placement of sensors around a Water distribution network, with the main goal of that placement being the effective and efficient monitoring of Disinfection By-Products. Beyond this goal, freedom was given to decide freely what the specific goal of this graduation project was. A literature review was conducted at first, to gain knowledge about how existence of similar problems, feasible approaches to those problems, as well as limitations and gaps of these studies.
A Meeting with collaborators in the project indicated a feasible challenge that had relevance to the H2OforAll project, which will be further explained in the following section.

3.2 Further definition of goal

H2OforAll states the following: ” DBPs Sensing, Analysis and Visualizations ” for WP2, Environmental Impacts Risk Assessment of DBPs and Prevention Measures Analysis” for WP3 and ”Assess main DBPs sources and fate by developing suitable analytical techniques for monitoring such compounds and

Specific	The goal is defined with a specific focus in mind, that being placement of sensors of different
Measurable	The BWSN challenge will be a ”measuring stick” against which the own research will be me
Achievable	Similar research to the goal has been conducted, modelling tools for this research already
Relevant	This goal adresses an identified gap within implementation of sensors under the H2OforAll
Time Bound	This goal is constrained to the GP Period of MOD 11 and 12

Table 2: SMART-assessment of proposed research goal

modelling their spreading through drinking systems ”as one of the objectives of the project [28] . During a meeting with previously mentioned collaborators in this project, which have backgrounds in water engineering and research about water engineering, an insight was gained into limitations that a real life case presented. A network, included with a DBP growth model (in the form of an .msx file) and the corresponding sensor placement was requested from the collaborators. The collaborators then provided a network, with an uncalibrated chlorine propagation, instead of DBP model, and a sensor placement that was entirely limited by the local populations' will to participate. This was combined with the statement made that the sensors placed were each going to be different. The limitations of real life-implementation
Having identified these shortcomings in the application of trial sensor network for network modelling, combined with the knowledge That sensor placement heavily influences the effectiveness and efficiency of DBP monitoring, and that it is entirely possible to model a WDN that is able to aid in the decision-making process of real-life implementations, gave the prompt to specify the goal of this GP to suit this specific case. The GP's specified Goal is: To optimize sensor placement to monitor the growth of DBPs in a WDN, utilizing a set of sensors comprising sensors of different qualities. With a performance that compares to the BWSN, Within the time defined by the Graduation project.

3.3 SMART-assessment of specified goal

The SMART assessment helps define whether a proposed goal is, Specific, Measurable, Achievable, Relevant and Time-bound, Below is an assessment of the GP's proposed goal. [31] the goal is repeated for clarity: To optimize sensor placement to monitor the growth of DBPs in a WDN, utilizing a set of sensors comprising sensors of different qualities. With a performance that compares to the BWSN, Within the time defined by the Graduation project.

3.4 SWOT Analysis:

Conducting a brief SWOT analysis [46]will reveal further why the choice has been made to specify the GP in this manner. Important to note is that usually, the goal of SWOT analyses is to assess a companies' strategic advantages and weaknesses, so that it can make informed decisions on what specific goals

Strengths - Team Oriented - Holistic Approach - Multidisciplinary
Opportunities - Monitoring of DBPs in a sensor network is a new topic without many established specialist

Table 3: SWOT analysis of the CreaTe student and the Graduation project

to pursue. In this case, the SWOT will assess me as a ”company” identifying my own strengths and weaknesses as a student Creative Technology, in the larger scope of this H2OforAll project. **Strenghts** The strength of a Creative Technology student lies in the fact that they are taught to work in context of larger teams, as can be seen, for example in the project in MOD8 of the CreaTe program. They have a Holistic design approach that considers more than only the technical aspects of a project, and seeks to connect with, and design for all stakeholders involved in a project. On top of this, CreaTe students carry a set of knowledge that is inherently Multidisciplinary, which allows them to adapt and develop feasible plans and prototypes.

Weaknesses The weakness of a Jack-of-all-trades is that they they are by definition not specialists. This might constrain the CreaTe student to applications that are not highly-specialised.

Opportunities The opportunity in the case of WDN modelling and sensor placement for specifically Disinfection-By-Product monitoring, is that this is a relatively new field of study, meaning that there are relatively few people researching this project, meaning that there are also relatively little specialists in the field. When this is the case, the multidisciplinarity of the CreaTe student will serve them well.

Threats Lack of knowledge and experience with water engineering, or implementation of water sensors in a network, might result in the case that a proposed solution gets Veto-ed, because the CreaTe student does not have deep insider knowledge, or this is poorly communicated to them.

4 Methods and Techniques

To assess whether a hybrid approach to sensor distribution along a water distribution can be better for DBP detection, the Optimal sensor placement problem shall be conducted for both sensors of equal and varying qualities.

4.1 Methodology

The Underlying goal of the Project is delivering the citizens of Coimbra with clean, potable DBP-free water.
that goal is large and unbelievably broad, so we will focus on the prediction and accurate monitoring of DBPs, by way of Optimal sensor placement. the question we want to answer in the end is as follows: can we, with a sensor set comprising of several different sensors, monitor the growth of DBPs in a water distribution with a performance that compares to the the BWSN?
in Order to approach this problem we will use the data gathered through the literature review, as well as from the EPA, WHO and CDC, to generate a growth model of the DBPs. here the DBPs that are most relevant (or "dangerous to human health") will be selected and modelled in the form of equations that take the relevant parameters (such as pH, Temperature, Chlorine Residual) into them, and comprise them together in a .msx file. After the DBP growth model has been developed, it will be integrated into EPANET and corresponding softwares. Here it will be used to simulate DBP growth and propagation throughout a network. The BWSN1 and BWSN2 networks are good candidates because of their moderate to high complexity and their already established presence as a good benchmark. The case and network of Coimbra will also be considered, however inclusion of them in this Graduation may be omitted, because of the sensitive matter and potential for misuse of this information.
The modelling tools, described in Chapter 2 will be used to implement different sensors, algorithms and objectives to assess their performance of optimally placing sensors. This will be done by first assessing their performance over "simple scenarios" where there are only single DBP growth outbreaks from a preset point, with a single objective (TD). Gradually over the course of the testing trials, the scenarios tested will increase in seriousness and difficulty, until it approaches disaster-like scenarios.
The results of each algorithm and its performance for the various objectives and scenarios shall be presented in a way, similar to the BWSN, where the sensor placement, the algorithm and their respective performance will be shown. below is a methodology in bullet points summarising the general approach to this challenge.

4.2 summarised plan

1. Create an .msx file for relevant parameters, which are derived from literature.. here, we will take the DBPs considered most relevant/harmful to

human beings.

2. With the .msx file being a model of DBP growth, we can apply it to a network and understand the propagation of DBPs throughout a WDN.

3. After applying these to a model in TEVA-SPOT[25] with a predetermined amount of sensors, the effectiveness of the algorithms in providing an optimal solution can be assessed.

4. The algorithms should first be tried with a single-objective optimization, Keeping in mind that TD, DL, VC and DC are among the most used objectives

5. Afterwards, Multi-objective problems should be considered, since they are trickier to assess. these will be trying different algorithms and MCDA-methods

6. Iteratively the ”DBP-Outbreak scenarios” should increase in severity.

7. Once there is confidence in the accuracy of the model and our sensor placement locations. different quality sensors should be tested. such as to see whether a hybrid approach can offer more than a current approach, where all the sensors are assumed to be the same.

8. This testing can approach the Coimbra case as much as possible, in the sense that the sensor set can be the exact sensor set that

9. at last a comparison of prices of implementations, by finding real life sensor cost and giving an assessment of how much placement in a network will save resources.

10. review the performance of each optimal placement configuration, and present in a manner that highlights the strengths and weaknesses of the placements

4.3 chosen scenarios

for optimal sensor placement to make sense, its effectiveness should be tested against a number of scenarios, which might be realistic cases that a WDN might encounter. These include, but are not limited to: spontaneous/ purposeful injection, multiple species,more injections than sensors,breaches of water quality regulations/legislation, where they result in Generation of DBPs, Infected sources

4.4 Selection of Algorithms

- Multiple placement algorithms should be tested: multiple algorithms will give us ground to compare them against each other, much as was done in the BWSN.

Week	Goal
1	
2	creation of .msx file
3	Single-objective optimisation for TD, VC, DL and DC
4	Multi-Objective Optimisation, starting with NSGA-II (because it has been used most)(incl. MCDA)
5	ˆ idem
6	Application of Multi-Objective algorithms to chosen (special) scenarios
7	Inclusion of Hybrid Quality sensors into system Inclusion of sensors with different qualities
8	(second) reviewing of collected simulations and data, repairs where needed.
9	finalising of report, as well as preparation of presentation
10	Final report and Presentation hand-in

Table 4: Graduation Project Roadmap

4.5 Tool selection

Tools, such as EPANET, EPANET-MSX , TEVA-SPOT, CHAMA, and the MATLAB toolkit will be utilised where the strengths of each tool will help understand a different aspect of the water distribution network.

4.6 Road-map

5 Specification

5.1 Scenario

to make sense of the project, a scenario will be proposed, which will then be taken as the inspiration for the proposed end product, as well as an explanation as why several choices have been made.
current sensor placement by water management does not consider placement of their sensors on a network, because water management could not persuade other involved water managers/parties to place sensors around the network. Aguas de coimbra resorted to asking several users of the WDN, such as schools or hospitals to ask whether they could place a sensor there. After incredible efforts the end they ended up with 3 locations to place auxiliary sensors, where the plan was to do rule-based placement of water analyzers.
water companies could not persuade the other parties to place sensors, and if they did, sensor placement would be not optimal, since they take a rule based approach, which is prone to not consider all the parameters involved. this would cause sensor placement in water distributions to be sub-optimal, because water engineers lack the knowledge of optimal placement, as well as the permissions for such placement. this results in confusion in where sensors should be placed, as well as a less convincing argument for other parties to collaborate with the current sensor placement.

5.2 idea proposal

in this GP a better approach to this strategic sensor placement problem will be presented: By having access to an encyclopedia of optimal sensor placements, Water network managers can decide where to place their sensors, according to a certain scenario or scenarios they are trying to avoid. This will serve the water distribution managers in two different ways: first, they will have a clearer, optimisation based approach, which considers all cases and mathematically chooses the best option.
secondly, the visualisations will be a powerful tool to make other parties involved aware of the current situations, as well as to assess what the effect is of different sensor placements around a network. It should be a tool to help the water network managers answer many "what if?" questions, that they might have when designing a network. they might decide to answer questions about addition of one or more sensors to a network, or optimisation for sensor placement for a very specific DBP. the idea is that it is an encyclopedia of sensors placements. this encyclopedia should include multiple simulations and configurations for sensor placement, as well as an explanation of its strong points, which will be explained through its performance metrics. such as TD, VC, PE, DC, Detection probability for a certain DBP. the encyclopedia should also prepare the user for several calamity scenarios, as well as propose network configurations that could be effective against these scenarios note that here, the goal is not to decide for the Water management organisation(s),

rather, it is provide them with several configurations and scenarios, along with performance metrics so that local water management organisations can make informed decisions about what they deem feasible in terms of resources, time and performance.
it is important to note that the proposal shall consider the placement of sensors across the entire network to be equally challenging, so as to not try to predict something that water management parties already know best, namely the placement of sensors with regards to human behaviour, accessibility of pipes and possible legislation/permissions.

from the *scenario* and *idea proposal* subsections above, we can draw several requirements that will be split in functional and non-functional requirements.

5.3 functional requirements

The functional requirements are defined as ”what the product/service does” by the Graduation Project manual. below is a description of goals that the end result of this project must reach in order to be called successful.
The end product must explain sensor placement in an understandable manner, such that a water network manager (such as Aguas de Coimbra), can make an informed decision about the placement of their sensors for DBP detection. it must also support the configurations with sufficient explanation, so that those without much knowledge about modelling can understand reasoning behind the different configurations.the end-product must include several different calamity scenarios where DBPs grow inside a water distribution network, along with several configurations that are tailored to protect against each scenario. These shall be presented in such a manner as to bring out the strong and weak points of each configuration as well as tradeoffs against other configurations.
These threat scenarios are analogous to real life and include the single objective optimizations for VC, DL, DC and PE, as well as multi-Objective optimisations for the detection of different families of DBPs. for the multi-objective optimisation MCDA analysis tools, as well as example applications of those shall be provided. These can then be later be used by the Water Network management, along with other parties to properly align their wishes and decide on what sensor configuration will be best suited for their network cases. in the end, the code, as well as several other example networks shall be included in this encyclopedia package.

5.4 non-functional requirements

The non-functional requirements are defined as ”how the product/service does it” by the Graduation Project manual. The following description includes the method and the reasoning behind inclusion/exclusion of parameters, such as to support the functional requirements of the project as best as can.

5.4.1 DBPs

the DBPs that are going to be measured are the 3 Families, of HANs, HAA and THMs since according to [30]they are most carcinogenic.

5.4.2 parameters

the main parameters that these growth models of DBPs will take in are Temperature, pH and Chlorine Dosage. these parameters are applied to the model in two different ways, in nodes with a favorable environment for the DBPs, these will have a higher likelihood to have DBPs be ”injected” in the system. the second manner in which these parameters will be impacting a system is that continuously favorable conditions will increase the ”injected” dose. the way they are implemented here is to ensure that they will be abstractions of families of DBPs that are easily tweaked/modified, as well as still provide an approximation of how DBPs grow in a network in a real life scenario.

5.4.3 scenarios

the scenarios are a set of (sometimes unlikely, but not unrealistic) threats that network can endure. they will be presented across multiple scenarios increasing in threat severity, i.e. single injections over a network spontaneous/ purposeful injection, more injections than sensors, breaches of water quality regulations/legislation, where they result in Generation of DBPs, Infected sources. The scenarios mostly decide severity of the Threat, as well as point out interesting effects that might occur in a network, that would be good to know for a Water management organisations.

5.4.4 algorithms

several algorithms shall be tested, to assess the success of each algorithm in finding optimal solutions, what we are most interested in here is to assess whether the Pareto fronts have a difference, or converge to the same level.
- **Coimbra base case.** the Coimbra base case placement of sensors shall be considered to be the base level, against which all other placements shall be waged against first. this is because it is the real life scenario, from which it becomes easy to spark ”What if?” questions. For example: ”What if the Coimbra Network used optimised sensor placement to minimise the Population Exposed to contaminated water?
- **Random placement** The inclusion of Random Placement across a water distribution shall have the function of proof that placement using the algorithms consistently beats random placement. It serves to assess both Coimbras current sensors placement configuration plan, as well as the configurations that both of the following algorithms will provide.
- **GRASP heuristic** the GRASP heuristic is an algorithm that will try to find global optima in a greedy manner, with a randomisation element(that is included to ensure it does not it gets stuck in local optima a lot of times)

- **NSGA-II** NSGA-II is a Genetic algorithm that tries to find global optima using an evolutionary approach, that is further explained in chapter 2. Most important to know is that this is a well known algorithm, well performing algorithm that is used in a vast amount of different problems.NSGA-II is used to assess whether the GRASP heuristic is in fact enough, as literature and tools state, or whether it can be consistently beaten by another algorithm. other optimisation algorithms shall be omitted, since inclusion of those will shift the research question towards the performance of different algorithms, instead of the application of an algorithms to optimise for DBP detection.

5.4.5 sensors

different sensor scenarios will also be tested. these include "abstracted" sensors that detect DBPs. this is the approach that has been followed in the literature that has been found so far. for sensor placement to be closer to real life, the different sensors that are being used in the case of the Coimbra network. are going to be implemented in the model. chlorine dosage, NOM composition, Temperature, pH, electrical conductivity, dissolved Oxygen and Oxidation-Reduction Potential (ORP). an approximation of these sensors as well as their threshold values shall be included in the model. an assessment of whether these perform similarly to the abstracted sensors will also be included.
Centrality Based Sensor Placement
By taking a Network Topology based approach, e.g placement of sensors around the central nodes of a network, based on beweenness centrality a rule-based heuristic, might be developed that still has moderate performance, yet does not need heavy computational resources to be calculated.

6 Realisation

6.1 Tools used

IN short, the tools that have been used are the following:

- WNTR and NetworkX have been used to extract the nodes with the highest centrality score. TEVA- SPOT has been used to calculate network sensor locations based on minimizing the average TD times Using the GRASP Heuristic.

- EPANET has been used to perform multiple 'sanity checks', to see whether the network looks fine and acts accordingly. Since EPANET does not have any -MSX capabilities, usage of these is solely done in The EPANET MATLAB TOOLKIT. which has also been used to do the Majority of calculation within this Graduation Project, as will be further explained in the following paragraph

6.2 Initial Test run

at first, an early test run, regarding the testing of sensor placement has been done, to gain an understanding of what is feasible, in terms of computation. As well as come up with an initial good plan for Assessing whether a strategy based purely on the score of betweenness centrality:
chosen is a network Net3, an example network, that is often used, and includes operation of pumps and reservoirs that are analogous to a real life system. first, the top N central nodes were calculated, by importing the network in python and using NetworkX to calculate the Betweenness Centrality. for the simulation, The EPANET MATLAB Toolkit was used, where the compounds moving through the network have been simulated trough an extension .msx file. the MSX file decides injection/contamination locations of various compounds.
Using the EPANET MATLAB TOOLKIT two realistic scenarios are simulated:

- **Scenario 1 - Contaminated sources** in scenario 1 the RIVER and LAKE nodes are sources of TOC. through them the TOC makes its way into the network. The TOC reacts with the Chlorine, that is coming from the Tank nodes, 1,2 and 3 respectively. Chlorine from the Tanks react with the TOC to create THMs, one of the major DBP families. As a sanity check, 3 Network visualisations are made, to confirm that the .msx file has been configured correctly

- **Scenario 2 - Leaks in the system**. in scenario 2 the 4 nodes have been selected to be sources of TOC. this is done with the intent to simulate leaks in a water distribution system, through which natural organic matter can make it into a water distribution system. as with Scenario 1, the tanks in Scenario 2 are also the sources of Chlorine.

to assess whether Centrality could be an effective measure of DBP detection, the 15 highest centrality nodes have been selected and a timeseries of the concentration of THMs at their nodes have been made. these have then been studied, in order to reach a conclusion of what needs to be done in the following iteration.

6.3 conclusion initial test run

In scenario 1, the Sensor placement based on betweenness centrality seemed to be an effective measure for detection of DBPs in this specific configuration, by having an early time of detection and relatively little variance in detection.
In scenario 2, the Sensor placement based on betweenness centrality seemed to be less effective for detection of DBPs. several nodes did not detect a high amount of DBPs. and those that did usually much later than in scenario 1. more trials of random injection locations have to be run to accurately assess whether random injection location is overall worse scoring.
further measures to investigate upon are whether this pattern extends in larger networks, and/or networks with a higher 'branchedness'.
The hypothesis for now is that networks with a higher 'branchedness' and contaminated Sources closer to the highest central nodes will score better in terms of betweenness-centrality-based sensor placement.
Based on the first test run, it has been decided to investigate further into Scenario 2, where leaks present themselves randomly throughout the network, since it is deemed the more challenging case for sensor placement, with an extra layer of uncertainty being generated as a result of the random injection locations. Important to note is that in addition to this random factor in sensor placement, there is still a growth model of THMs present in the simulation.
to discern whether networks of a different topology classification higher branchedness make a difference, Three Networks that are classified as Branch, Loop, and Grid forms, respectively, shall be selected and simulated.
Lastly, these simulations for the networks will have to simulated many time, such that statistical inferences can be made based on the simulations.

6.4 The Testing procedure:

6.5 Network Selection

Select a Branch, Loop and Grid Network from a database.

- These have been seleted from a database from a WDN systems research database provided by the University of Kentucky. [50] . Since it is a moderately sized database which has documentation on each of the networks that are provided. From this Database a GRID, LOOP and BRANCH network have been selected, these being the networks Net3 [3], Modena[2] and PA2[1], respectively. During the selection of the networks, medium-small networks were chosen , to attempt to create similar scenarios for the three networks.

6.6 Calculation of Central Nodes

To Discern which nodes are the top 15 Central Nodes, The use of 2 python packages - WNTR and NetworkX have been used. WNTR is mainly used because of its compatibility with EPANET as well as NetworkX. This prevents confusion when attempting to reference the same in MATLAB. NetworkX was used to calculate the betweenness centrality score of each node within the network By simply using:

```
    betweenness_centrality = nx.betweenness_centrality(G)
```

the Centrality for each node in the network would be calculated. furthermore, using:

```
N_sensors = [15]

for n in range(0,len(N_sensors)):
    N = N_sensors[n]
    topNcentrality = sorted(betweenness_centrality, key=betweenness_centrality.get, reverse=
    print(topNcentrality)
```

The list of nodes could be sorted by the central nodes, and printed to the console. finally, a plot was made, with the top 15 nodes higlighted , to visually 'confirm' the nodes being central to the network and as a visual aid. at last, the selected nodes are exported to MATLAB for further use.

6.7 Sensor locations using TEVA-SPOT

To have a clearer answer to the performance of using betweenness centrality as a strategy for detection of DBPs in a water distribution system, the centrality strategy has to be measure against a commonly used method of sensor placement. The chosen method to compare against is using TEVA-SPOT and a GRASP Heuristic, with Time to Detection as its main metric to be minimized.
for each of the networks, the Network is loaded into The TEVA-SPOT-GUI, after first assigning an Ensemble and Collection folder for the network to be placed in. Next, Running step-by-step through the Execution Control tab, the network is prepared for simulation and sensor placement. The network simulation time has been set to 48 hours, The injection of THMs into the network using 4 random placements throughout the network.
for the Sensor placement, 15 sensors with a detection limit of 1 μg, are placed around the network, using the GRASP heuristic as the optimisation method of placement of sensors in the network. after executing all the necessary steps. TEVA-SPOT, provides A visualisation of the networks as well as several tables containing scores and performance metrics, as well as the Network nodes where sensor placement is calculated to have the minimal average time to detection. The selected nodes are then exported to MATLAB for further use.

6.8 Creation of DBP growth model

As a basis for the growth model, The "DBPRisk MATLAB library, also developed by Eliades et al. has been used as a basis, as it contains a growth model for THMs, based on chlorine, TOC, Water age, pH and Temperature. This has been taken as a template, and altered in such a way to be able to make use of it in MATLAB EPANET TOOLKIT, as well for it to be able to be compatible with multiple networks, without manual altering after each simulation round. removing any "predefined" sensor location, the growth model could be used on different networks at will, provided that the injection locations are correctly referencing the nodes of the corresponding Network. this referencing was either done by manually altering the .msx files [SOURCES] and [QUALITY section] or by a MATLAB script to rewrite the entire file each time it is loaded, making sure this happens **After** loading the network(.inp) file into the toolkit.

6.9 Sensor detection in MATLAB

To simulate sensor detection within MATLAB, A simulation of the hydraulics of the network (.inp) , as well the (bio-)chemical interactions within the network(.msx) is run. This is Followed by the plotting of the concentration of THMs present within a selected within the network throughout the simulation and checking when the concentration of said chemical crosses a preset threshold value. In the case that this happens, the time at which this happens is saved. This process is repeated for each of the 15 nodes, for each simulation. This is finally outputted as the first detection times for each of the 15 nodes.

6.10 Randomised placement in MATLAB

To discern whether strategic sensor placement makes any difference to detection times within a network, it must be assessed against a case where there is no use of any strategy. Random Placement of Network nodes throughout the network simulate this case. by using the following lines of code:

```
  node_count = d.getNodeCount
initrandsensorArray = randi([1, node_count], 1, 15)
sensor_index = initrandsensorArray
```

Random Sensor placement configurations are made, by considering the amount of nodes present in the Network of the Simulation, after which the code selects 15 indices randomly. These 15 nodes will then be assessed as if it was a sensor placement strategy, meaning that the detection times will be recorded and outputted.
similarly, when using this method within a for-loop, many random sensors placements can be generated. This technique will be used and explained promptly

	GRID (Net3)	BRANCH (PA2)	LOOP (Modena)
Central Nodes C	Case 1	Case 5	Case 9
TEVA -SPOT T	Case 2	Case 6	Case 10
60 Random Placements R60	Case 3	Case 7	Case 11
1 Random Placement R1	Case 4	Case 8	Case 12

Table 5: Name of cases

6.11 Full Simulation, Placement and Sensor Detection in MATLAB

now that the individual steps have been explained, we can combine the individual steps and explain how Sensor placement has been selected, and assessed.
Firstly, The sensor placement strategies, by TEVA-SPOT and NetworkX are imported into MATLAB, referenced by their index, combined with a hydraulic model (.inp-file) of the corresponding file. Next, a hydraulic model (.msx file) is written, to simulate leaks around the network, done **after** the initialisation of the.inp file, to ensure successful injection of TOC into the WDN, meaning that Nodes, Sources and Tanks are being correctly referenced for injection . The injection is set to have 4 injection locations of TOC, 3 having a concentration of 0.1, and 1 having a concentration of 0.2.
Once the file has been written and 4 random injection locations have been selected, A Hydraulic and biochemical simulation is run, the 15 Nodes are assessed over their quality and detection times of THMs are outputted for each of the nodes. the detection times for the 15 selected nodes is outputted.
this is process is repeated for 60 iterations, i.e. 60 different TOC intrusion models, where TOC makes its way into a network, resulting into growth of THMs into the Water Distribution Network. for the Networks, Net3, Modena and PA2 this process is repeated 4 times, for each of the different sensor configurations. that means that in total 3 Networks x 4 cases x 60 Iterations are run. The number of iterations because it strikes a good balance between computational time and ability to generate statistical inferences. for each case that is run the iterations are Exported to excel, where they will be evaluated. the Data for each of the different cases is a block of detection times that is 60 columns and 15 Rows, These represent the each of the iterations and the placement of each of the 15 sensors

6.12 Evaluation of Simulations in Excel

After all the results have been Exported Excel, The results were organised sorted by Network, and each specific Case for the Network has been given a case number. After the data had been sorted by Cases, The Average-, Minimum-, and Maximum detection times are Determined, along with the number of failed detections. These metrics are calculated For each of the iterations, as well as each of the 15 selected nodes. is collected, after this, for each case The following Calculations are made in Excel:

- **for each case 1-12:**ANOVA Between all the selected nodes. In order to determine whether centrality has an effect on Time to detection. in this calculation all non-detection entries would skew the results, therefore all the empty cells(Nondetection values) have been set to equal 1.5 Times the Simulation duration of 2 days.

- **for each case 1-12:** ANOVA Between all the selected nodes. In order to determine whether centrality has an effect on Time to detection. in this calculation all non-detection entries would skew the results, therefore all the empty cells(Nondetection values) have been set to equal 1.5 Times the Simulation duration of 2 days. **for each of the Networks:** ANOVA using # of Failed detections between each of the sensor placement strategies. (e.g for Net3, ANOVA between Case 1-4)

- **for cases 1, 5, 9:** Linear Regression for the ordered set of 15 central nodes, and each of their respective times,in order to discern whether centrality score is correlated to time to detection.for cases 1, 5, 9: Linear Regression for the ordered set of 15 central nodes, and each of their respective percentages of failed detections, in order to discern whether centrality score is correlated to failures to detect.

- **Between the cases 1 and 2, 5 and 6 , 9 and 10:** A t-test to determine whether The Centrality strategy and TEVA-SPOT Strategy have significantly different average detection times for THMs

After calculating all of the above, evaluation of the gathered data and the method to obtain so can be assessed

7 Evaluation

7.1 NET 3 - GRID - CASES 1-4

7.1.1 Network Sensor Distribution

For Central Nodes: (fig.1) Central nodes are clustered together, chaining up together **For TEVA-SPOT**(fig.2) Fairly well distributed throughout the network. few nodes that connect to each other. **ANOVA BETWEEN METHODS** (fig.19) The P-Value is lower than 5% for Case 1-4 , meaning that Placement strategy has a significant effect for Time to detection. **AVERAGE DETECTION TIMES** for cases 1-4, Centrality strategy seems to be not beneficial, being beaten in time to detection by Random Placement of a single sensor set. **CORRELATION BETWEEN CENTRALITY AND AVERAGE DETECTION TIME**(fig.4) Centrality seems to be negatively affecting, detection time, having a longer detection time as more central nodes are approached **Correlation between Centrality and percentage of failed detections:**(fig.6) Centrality seems to be positively influencing the of failed detections, meaning that it is less likely to miss a harmful pathogen as placement is done on more central nodes **Statistically significant difference** (fig.22)(between Case 1 and 2) performing a two-tailed T-test on the two sensor detection samples the P-value is smaller than meaning that there is a statistically significant difference between the two samples of detection times.

7.2 PA2 - BRANCH - CASE 5-8

For Central Nodes: (fig.7)Central nodes are clustered together, Chaining together really much, leaving only room for one node in between **For TEVA-SPOT** (fig.8)Fairly well distributed throughout the network. few nodes that connect to each other., bottom side less covered. **ANOVA BETWEEN METHODS** (fig.20)with a P-Value lower than 0.05 (namely 0.007) for Case 5-8 , it can be stated that Placement strategy has a significant effect for Time to detection. **AVERAGE DETECTION TIMES**(fig.9) for cases 5-8, the Centrality placement strategy seems to be the most beneficial for minimizing average time to detection, scoring faster times than the TEVA-SPOT sensor placement **number of failed detections**(fig.11) The Centrality based placement strategy has lowest mean, but on par for the biggest variance in with the R60 method **CORRELATION BETWEEN CENTRALITY AND AVERAGE DETECTION TIME**(fig.10)Centrality baseds sensor placement on Network Pa2, seems to correlate with detection time in a positive manner, meaning that the time to detect is faster as more central nodes are picked. **Correlation between Centrality and percentage of failed detections:**(fig.12) the trendline between Centrality and percentage of failed detections top 10 Central nodes is completely flat, indicating that at least for the top 10 nodes, no, relation between Centrality and missed observations exist **Statistically significant difference** (fig.23) (between Case 5 and 6) performing a two-tailed T-test on the two sensor detection samples the P-value is 0.023, meaning that there is a statistically significant difference between the two samples of detection times, (but barely so)

7.3 LOOP - Modena - Case 9-12

For Central Nodes:(fig.13) The Central node strategy does not look Central at all, instead, it is a string of nodes off to the side. **For TEVA-SPOT** (fig.14)Sensor Placement by TEVA-SPOT looks uniform and well distributed throughout the network. **ANOVA BETWEEN METHODS**(fig.21) running an ANOVA TEST between all the Sensor Placement strategies returns a that is far smaller than 0.000026, indicating that most certainly the Sensor placement Strategy makes a difference in Sensor Placement. **AVERAGE DETECTION TIMES** (fig.9-12) The average detection times of this network shall be left inconclusive, since the amount of detection is not enough to make any inference that has any statistical power. The same goes for the **number of faileddetections** (fig.17) **CORRELATION BETWEEN CENTRALITY AND AVERAGE DETECTION TIME**(fig.16) there seems to be a very slight correlation between the Central nodes and the average detection times, although this can not be confirmed with any certainty, because of the lack of actual detection times. **Correlation between Centrality and percentage of failed detections:** (fig.18) **Statistically significant difference** (fig.24)

8 Discussion

8.1 Assessment

8.2 Summary

In this Graduation Project, A method to Assess whether Network Centrality is a feasible Strategy for Sensor placement, with regards to detection of DBPs is proposed and tested. It includes the usage of multiple modelling tools, to gather the Central Nodes, Create a hydraulic and Water Quality model, Simulate the results and gather insights from it.
The results Imply that a Pure Betweenness Centrality-Based strategy is not guaranteed to yield good results, but that with a slightly altered approach it may be a good method to find a sensor set that strikes a good balance between a speedy detection and a high detection likelihood detection of THMs, without the need for Optimization-based strategies that require a significant amount of modelling knowledge.
These results could impact how decision makers handle Sensor placement problems, as this method might provide a method to do a hybrid, optimisation/rule based approach. as that is where the strength of this method lies. It is not an over-the-top-sensor placement algorithm with remarkable performance, rather it is a method for a user to introduce themselves to WDN modelling. A stepping stone with which the user can quickly get themselves acquainted with the topic, as well as generate sufficiently advanced sensor placement.

8.2.1 Multi-disciplinarity/specialisation

ecause, WDN Modelling is a multi-disciplinary practice that requires people from different backgrounds, not being well versed in water engineering, it is without doubt that choices will be made that are suboptimal.Whether that be in choice of placement strategy, tools, modelling, research. It is therefore then also a weakness of the program itself to not be able to close that gap of vast engineering knowledge, or substitute it for a tool.
Further study into development of this Strategy for sensor placement could be to see whether a slightly altered version of the Centrality based sensor placement strategy might yield good results. Also interesting would be to try to develop this tool further, as a teaching aid for laymen regarding Water Distribution Modelling, preferably with addition of an optimisation-based WDN Modelling, since both the proposed method and optimisation share a similarities. For example, the selection of random nodes, and assigning a score based on metrics decided upon by the Modeller are similar.

9 Conclusion

In this Graduation Project the subject of Water Distribution modelling and many surrounding matters have been explored. We've started by being introduced by the topic of DBPs, that are recently found to be very harmful to humans, to the problem of sensor placement, general modern ways to approach said problem, from there many and algorithms and methods of placement have been learned about. In essence it has been discovered how much effort has been going into protecting our water systems and our population, by maintaining our water system. A new respect has been gained, through learning about a single one of the many aspects that come with water engineering.

10 Bibliography

References

[1] *"02 PA 1" by Dominic L. Boccelli.* URL: https://uknowledge.uky.edu/wdst_us/2/.

[2] *"05 Modena" by Ashley Hall.* URL: https://uknowledge.uky.edu/wdst_international/5/.

[3] *"06 EPANET Net 3" by Lew A. Rossman.* URL: https://uknowledge.uky.edu/wdst_manuals/2/.

[4] *(PDF) A Practical Guide to Multi-Criteria Analysis.* URL: https://www.researchgate.net/publication/358131153_A_Practical_Guide_to_Multi-Criteria_Analysis.

[5] Oluwaseye S. Adedoja et al. "A state-of-the-art review of an optimal sensor placement for contaminant warning system in a water distribution network". In: *Urban Water Journal* 15.10 (Nov. 2018), pp. 985–1000. ISSN: 17449006. DOI: 10.1080/1573062X.2019.1597378.

[6] Sabtain Ahmad et al. "Sustainable Environmental Monitoring via Energy and Information Efficient Multinode Placement". In: *IEEE Internet of Things Journal* 10.24 (Dec. 2023), pp. 22065–22079. ISSN: 23274662. DOI: 10.1109/JIOT.2023.3303124.

[7] Mohamed Ateia, Haoran Wei, and Silvana Andreescu. "Sensors for Emerging Water Contaminants: Overcoming Roadblocks to Innovation". In: *Environmental Science and Technology* 58.6 (Feb. 2024), pp. 2636–2651. ISSN: 15205851. DOI: 10.1021/ACS.EST.3C09889. URL: https://www.epa.gov/water-research/water-sensors-toolbox.

[8] Majid Behzadian et al. "PROMETHEE: A comprehensive literature review on methodologies and applications". In: *European Journal of Operational Research* 200.1 (Jan. 2010), pp. 198–215. ISSN: 0377-2217. DOI: 10.1016/J.EJOR.2009.01.021.

[9] Jean-Pierre Brans and Bertrand Mareschal. "Promethee Methods". In: *International Series in Operations Research and Management Science* 78 (2005), pp. 163–186. ISSN: 08848289. DOI: 10.1007/0-387-23081-5{_}5. URL: https://link.springer.com/chapter/10.1007/0-387-23081-5_5.

[10] Bruno Brentan et al. "Water Quality Sensor Placement: A Multi-Objective and Multi-Criteria Approach". In: *Water Resources Management* 35.1 (Jan. 2021), pp. 225–241. ISSN: 15731650. DOI: 10.1007/S11269-020-02720-3/FIGURES/4. URL: https://link.springer.com/article/10.1007/s11269-020-02720-3.

[11] Myrna V. Casillas, Luis E. Garza-Castañón, and Vicenç Puig. "Sensor Placement for Leak Location in Water Distribution Networks using the Leak Signature Space". In: *IFAC-PapersOnLine* 48.21 (Jan. 2015), pp. 214–219. ISSN: 2405-8963. DOI: 10.1016/J.IFACOL.2015.09.530.

[12] Myrna V. Casillas et al. "Optimal Sensor Placement for Leak Location in Water Distribution Networks Using Genetic Algorithms". In: *Sensors 2013, Vol. 13, Pages 14984-15005* 13.11 (Nov. 2013), pp. 14984–15005. ISSN: 1424-8220. DOI: 10.3390/S131114984. URL: https://www.mdpi.com/1424-8220/13/11/14984/htm%20https://www.mdpi.com/1424-8220/13/11/14984.

[13] Liuhao Chen et al. "An ordered clustering algorithm based on K-means and the PROMETHEE method". In: *International Journal of Machine Learning and Cybernetics* 6.9 (Nov. 2016), pp. 917–926. ISSN: 1868-8071. DOI: 10.1007/S13042-016-0617-9. URL: https://www.infona.pl//resource/bwmeta1.element.springer-doi-10_1007-S13042-016-0617-9.

[14] Robert M. Clark et al. "Measuring and Modeling Chlorine Propagation in Water Distribution Systems". In: *Journal of Water Resources Planning and Management* 120.6 (Nov. 1994), pp. 871–887. ISSN: 0733-9496. DOI: 10.1061/(ASCE)0733-9496(1994)120:6(871). URL: https://ascelibrary.org/doi/abs/10.1061/%28ASCE%290733-9496%281994%29120%3A6%28871%29%20https://ascelibrary.org/doi/10.1061/%28ASCE%290733-9496%281994%29120%3A6%28871%29.

[15] *CYENS - H2OforAll*. URL: https://www.cyens.org.cy/en-gb/media/news/h2oforall/.

[16] Kalyanmoy Deb et al. "A fast and elitist multiobjective genetic algorithm: NSGA-II". In: *IEEE Transactions on Evolutionary Computation* 6.2 (Apr. 2002), pp. 182–197. ISSN: 1089778X. DOI: 10.1109/4235.996017.

[17] *Decision Deck*. URL: https://www.decision-deck.org/project/.

[18] Kegong Diao et al. "Sensor placement in water distribution networks using centrality-guided multi-objective optimisation". In: *Journal of Hydroinformatics* 25.6 (Nov. 2023), pp. 2291–2303. ISSN: 1464-7141. DOI: 10.2166/HYDRO.2023.057. URL: http://iwaponline.com/jh/article-pdf/25/6/2291/1334934/jh0252291.pdf.

[19] *Drinking Water Distribution Systems — US EPA*. URL: https://www.epa.gov/dwsixyearreview/drinking-water-distribution-systems.

[20] *EPANET — US EPA*. URL: https://www.epa.gov/water-research/epanet.

[21] Thomas A. Feo and Mauricio G.C. Resende. "Greedy Randomized Adaptive Search Procedures". In: *Journal of Global Optimization* 6.2 (Mar. 1995), pp. 109–133. ISSN: 09255001. DOI: 10.1007/BF01096763/METRICS. URL: https://link.springer.com/article/10.1007/BF01096763.

[22] Caspar V C Geelen et al. “Optimal Sensor Placement in Hydraulic Conduit Networks: A State-Space Approach”. In: *Water 2021, Vol. 13, Page 3105* 13.21 (Nov. 2021), p. 3105. ISSN: 2073-4441. DOI: 10.3390/W13213105. URL: https://www.mdpi.com/2073-4441/13/21/3105/htm%20https://www.mdpi.com/2073-4441/13/21/3105.

[23] *GitHub - sandialabs/chama: Python package for sensor placement optimization*. URL: https://github.com/sandialabs/chama.

[24] *GitHub - USEPA/EPANETMSX*. URL: https://github.com/USEPA/EPANETMSX.

[25] *GitHub - USEPA/TEVA-SPOT-Toolkit*. URL: https://github.com/USEPA/TEVA-SPOT-Toolkit.

[26] *GitHub - USEPA/WNTR: An EPANET compatible python package to simulate and analyze water distribution networks under disaster scenarios.* URL: https://github.com/USEPA/WNTR.

[27] Kannan Govindan and Martin Brandt Jepsen. “ELECTRE: A comprehensive literature review on methodologies and applications”. In: *European Journal of Operational Research* 250.1 (Apr. 2016), pp. 1–29. ISSN: 0377-2217. DOI: 10.1016/J.EJOR.2015.07.019.

[28] *H2oforall*. URL: https://h2oforall.eu/.

[29] *H2OforAll - Wetsus*. URL: https://www.wetsus.nl/project/h2oforall/.

[30] Kamilla M.S. Hansen et al. “Effect of pH on the formation of disinfection byproducts in swimming pool water – Is less THM better?” In: *Water Research* 46.19 (Dec. 2012), pp. 6399–6409. ISSN: 0043-1354. DOI: 10.1016/J.WATRES.2012.09.008.

[31] *How to write SMART goals (with examples)*. URL: https://www.atlassian.com/blog/productivity/how-to-write-smart-goals.

[32] Zukang Hu et al. “Optimal sensor placement for contamination identification in water distribution system considering contamination probability variations”. In: *Computers & Chemical Engineering* 153 (Oct. 2021), p. 107404. ISSN: 0098-1354. DOI: 10.1016/J.COMPCHEMENG.2021.107404.

[33] Xin Jin and Jiawei Han. “K-Means Clustering”. In: *Encyclopedia of Machine Learning* (2011), pp. 563–564. DOI: 10.1007/978-0-387-30164-8{_}425. URL: https://link.springer.com/referenceworkentry/10.1007/978-0-387-30164-8_425.

[34] J. Kennedy and R. Eberhart. “Particle swarm optimization”. In: *Proceedings of ICNN’95 - International Conference on Neural Networks* 4 (), pp. 1942–1948. DOI: 10.1109/ICNN.1995.488968. URL: http://ieeexplore.ieee.org/document/488968/.

[35] Katherine A. Klise, Bethany L. Nicholson, and Carl Damon Laird. "Sensor Placement Optimization using Chama". In: (Oct. 2017). DOI: 10.2172/1405271. URL: https://www.osti.gov/servlets/purl/1405271/.

[36] Katherine A. Klise et al. "A software framework for assessing the resilience of drinking water systems to disasters with an example earthquake case study". In: *Environmental Modelling and Software* 95 (2017), pp. 420–431. ISSN: 13648152. DOI: 10.1016/J.ENVSOFT.2017.06.022.

[37] Zilin Li et al. "Gated graph neural networks for identifying contamination sources in water distribution systems". In: *Journal of Environmental Management* 351 (Feb. 2024). ISSN: 10958630. DOI: 10.1016/J.JENVMAN.2023.119806.

[38] Jaivik Mankad, Balasubramaniam Natarajan, and Babji Srinivasan. "Integrated approach for optimal sensor placement and state estimation: A case study on water distribution networks". In: *ISA Transactions* 123 (Apr. 2022), pp. 272–285. ISSN: 0019-0578. DOI: 10.1016/J.ISATRA.2021.06.004.

[39] *MATLAB*. URL: https://www.mathworks.com/products/matlab.html.

[40] Denis Nono, Phillimon T Odirile, and Bhagabat P Parida. "Assessment of probable causes of chlorine decay in water distribution systems of gaborone city, Botswana". In: *Online) = Water SA* 45.2 (2019). DOI: 10.4314/wsa.v45i2.05. URL: https://doi.org/10.4314/wsa.v45i2.05Availableonwebsitehttp://www.wrc.org..

[41] Avi Ostfeld et al. "The Battle of the Water Sensor Networks (BWSN): A Design Challenge for Engineers and Algorithms". In: *Journal of Water Resources Planning and Management* 134.6 (Nov. 2008), pp. 556–568. ISSN: 0733-9496. DOI: 10.1061/(ASCE)0733-9496(2008)134:6(556). URL: https://www.researchgate.net/publication/221936303_The_Battle_of_the_Water_Sensor_Networks_BWSN_A_Design_Challenge_for_Engineers_and_Algorithms.

[42] *Outranking Methods in MCDA: Pros and Cons*. URL: https://www.linkedin.com/advice/0/what-advantages-disadvantages-using-outranking-1e.

[43] Andrea Ponti, Antonio Candelieri, and Francesco Archetti. "A New Evolutionary Approach to Optimal Sensor Placement in Water Distribution Networks". In: *Water 2021, Vol. 13, Page 1625* 13.12 (June 2021), p. 1625. ISSN: 2073-4441. DOI: 10.3390/W13121625. URL: https://www.mdpi.com/2073-4441/13/12/1625/htm%20https://www.mdpi.com/2073-4441/13/12/1625.

[44] *splace-toolkit/EPANET-MATLAB-Toolkit/epanet.m at master · KIOS-Research/splace-toolkit · GitHub*. URL: https://github.com/KIOS-Research/splace-toolkit/blob/master/EPANET-MATLAB-Toolkit/epanet.m.

[45] *Sustainable Development Goals — United Nations Development Programme*. URL: https://www.undp.org/sustainable-development-goals.

[46] *SWOT Analysis With SWOT Templates and Examples !* URL: https://www.mindtools.com/amtbj63/swot-analysis.

[47] *Taveiro, Ameal e Arzila - Wikipedia*. URL: https://en.wikipedia.org/wiki/Taveiro,_Ameal_e_Arzila.

[48] *Universidade de Coimbra*. URL: https://www.uc.pt/.

[49] *Water Disinfection with Chlorine and Chloramine — Public Water Systems — Drinking Water — Healthy Water — CDC*. URL: https://www.cdc.gov/healthywater/drinking/public/water_disinfection.html.

[50] *Water Distribution System Research Database, University of Kentucky — University of Kentucky Research — UKnowledge*. URL: https://uknowledge.uky.edu/wdsrd/.

[51] WHO. "Chemical safety of drinking-water". In: (2007), p. 160. URL: https://www.who.int/water_sanitation_health/publications/dwchem_safety/en/.

[52] WHO. "Water Safety in Distribution Systems". In: *World Health Organization* (2014), p. 157.

[53] World Health Organization. "Guidelines for drinking-water quality: Nitrite and Nitrate". In: (2022), pp. 398–403.

[54] Tao Wu et al. "Traditional methods and biosensors for detecting disinfection by-products in water: A review". In: *Environmental Research* 237 (Nov. 2023), p. 116935. ISSN: 0013-9351. DOI: 10.1016/J.ENVRES.2023.116935.

[55] Wei Zhang et al. "Recent AdVances on Electrochemical Sensors for the Detection of Organic Disinfection Byproducts in Water". In: *ACS Sensors* 4.5 (May 2019), pp. 1138–1150. ISSN: 23793694. DOI: 10.1021/ACSSENSORS.9B00272/ASSET/IMAGES/MEDIUM/SE-2019-00272N{_}0007.GIF. URL: https://pubs.acs.org/doi/abs/10.1021/acssensors.9b00272.

[56] Yue Zhao et al. "New formulation and optimization methods for water sensor placement". In: *Environmental Modelling & Software* 76 (Feb. 2016), pp. 128–136. ISSN: 1364-8152. DOI: 10.1016/J.ENVSOFT.2015.10.030.

11 Appendix

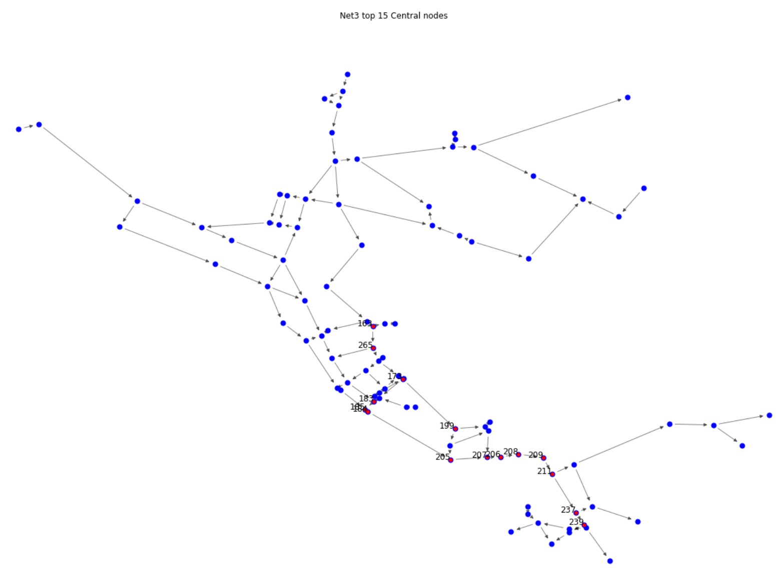

Figure 1: 15 Central Nodes of Net3

11.1 list of figures

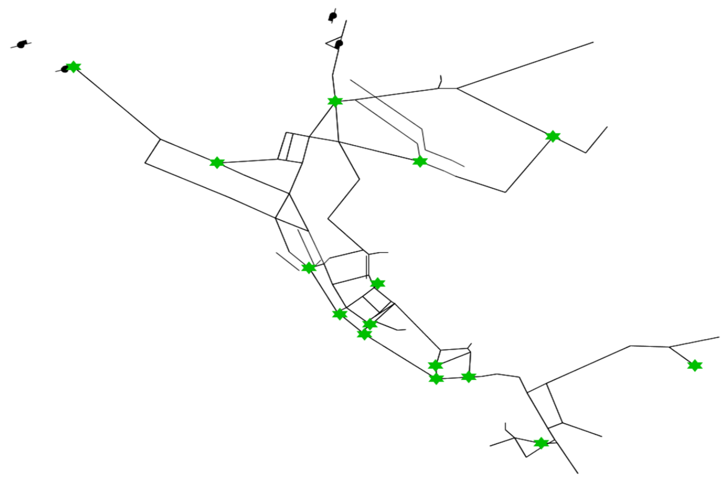

Figure 2: 15 Nodes minimizing TD in NET3

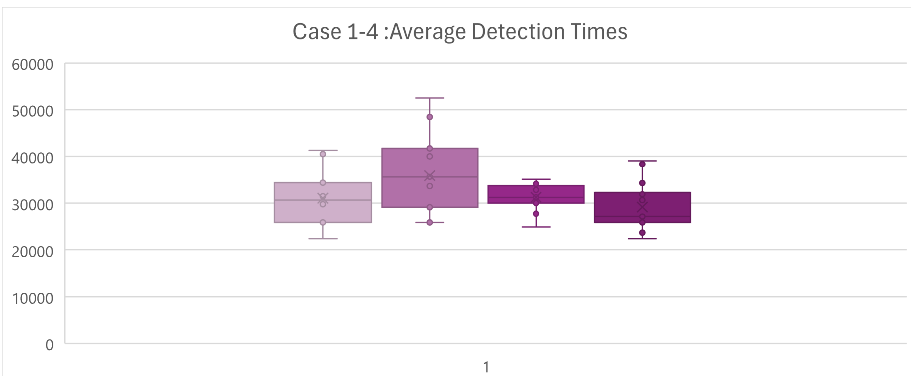

Figure 3: Average detection times of 1-4 from left to right

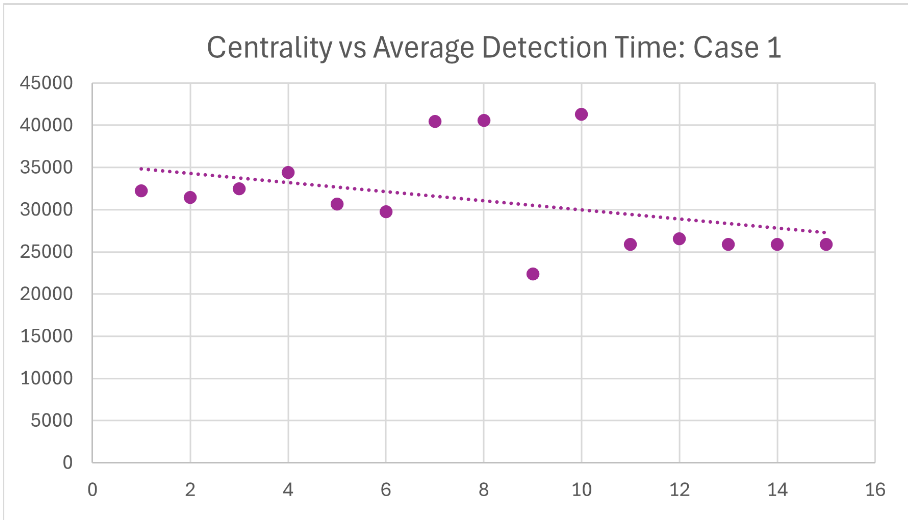

Figure 4: Higest 15 Centrality nose against detection time

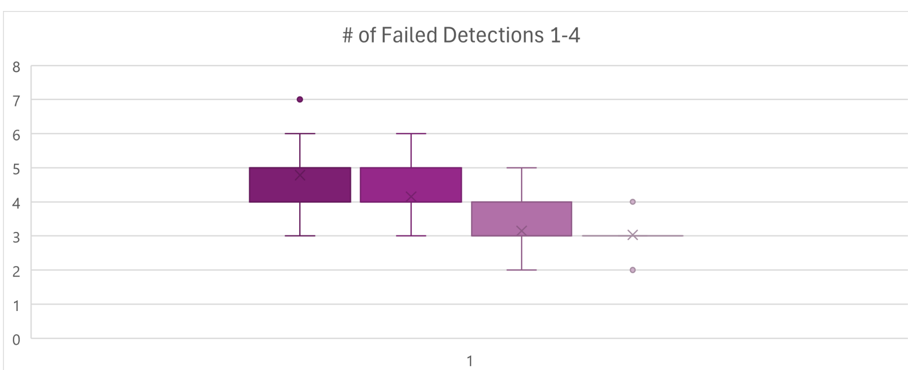

Figure 5: Number of failed detections per strategy (1-4 from left to right)

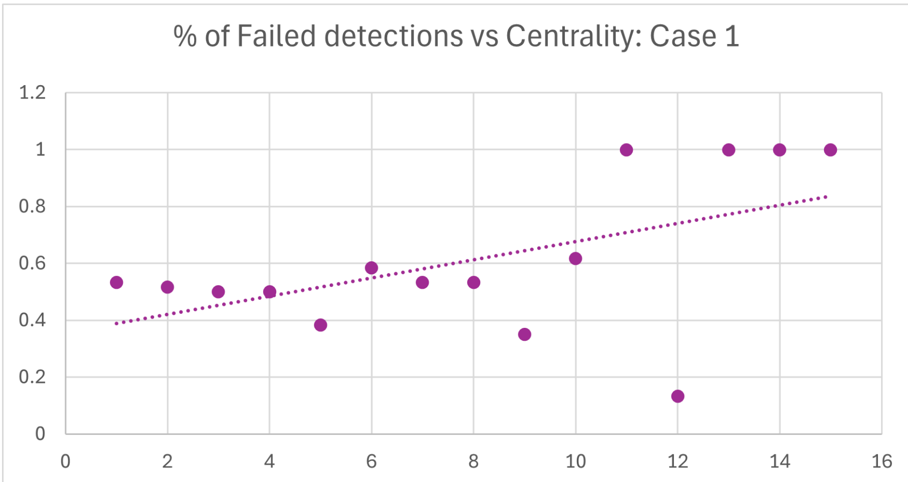

Figure 6: Percentage of Failed Detections on the top 15 Central Nodes

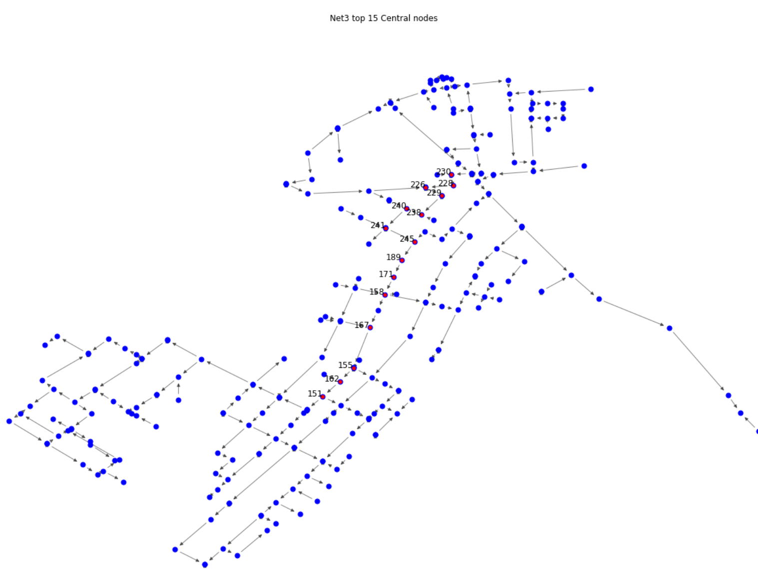

Figure 7: 15 Central Nodes of PA2

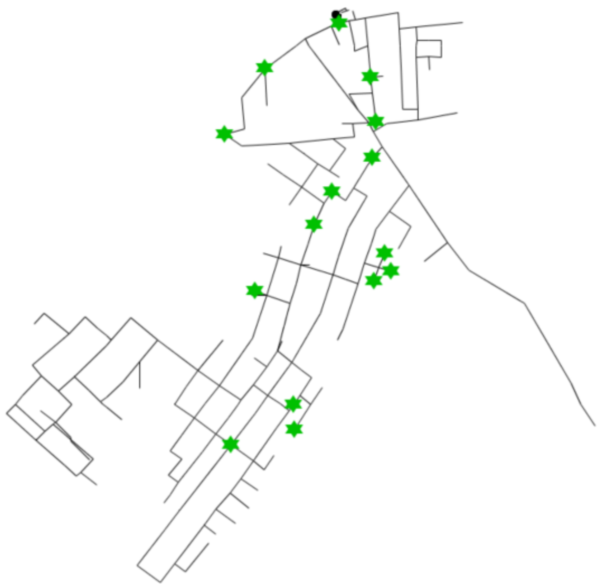

Figure 8: 15 Nodes minimizing TD in PA2

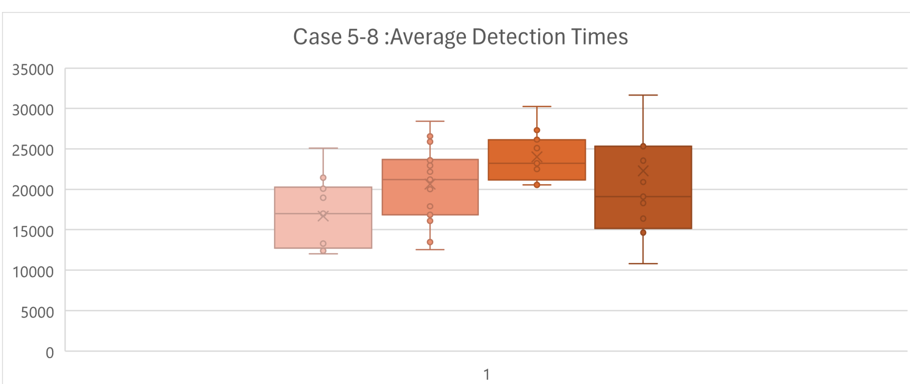

Figure 9: avg. det times for Cases 5-8 from left to right

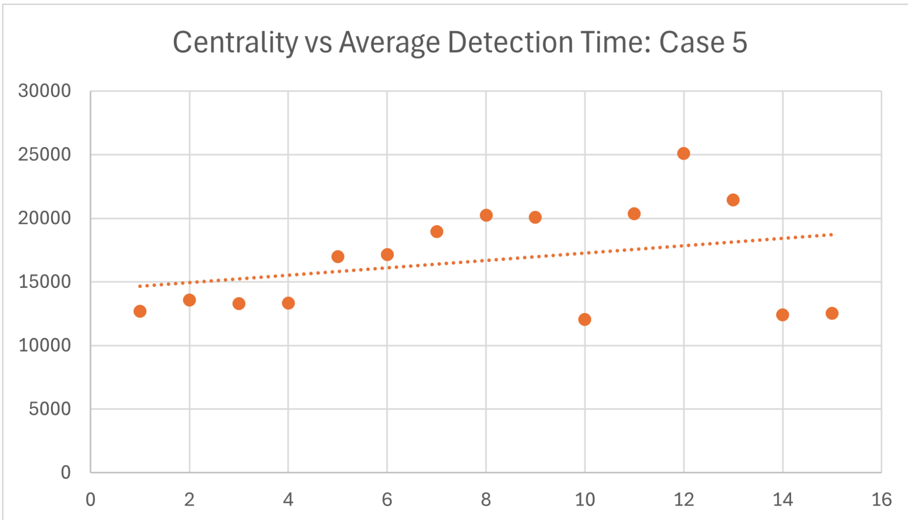

Figure 10: correlation between central nodes and avg detection time

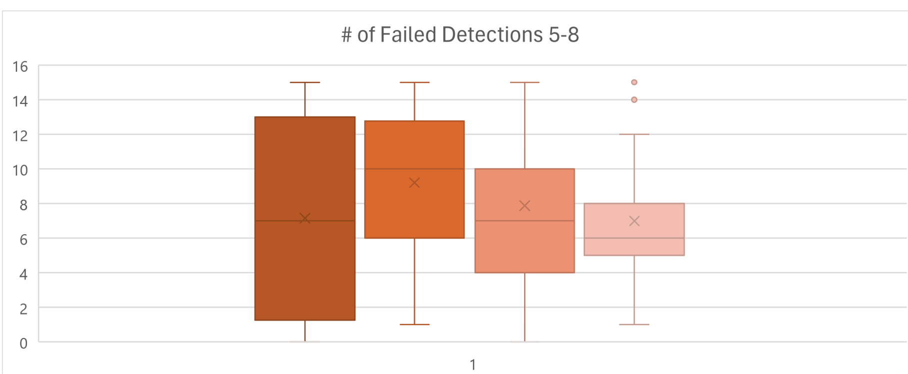

Figure 11: of failed detections for CASES 5-8, from left to right

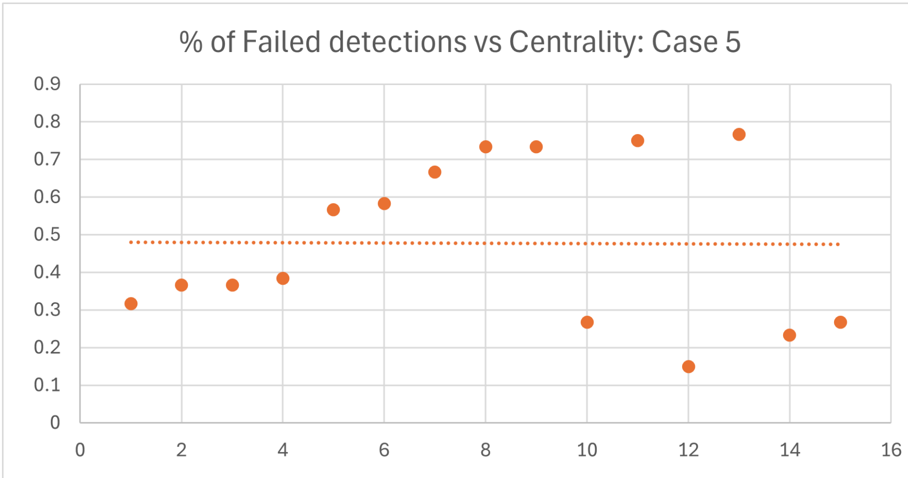

Figure 12: % of failed detection vs Central nodes

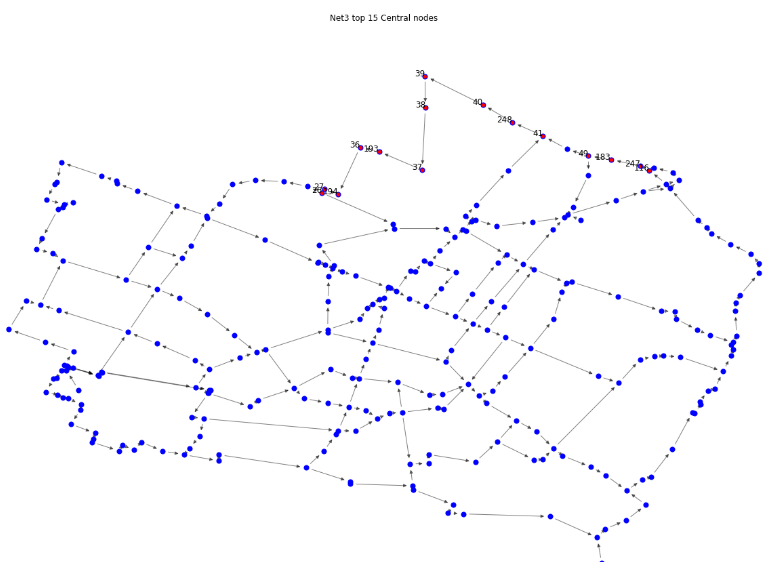

Figure 13: 15 Central Nodes of Modena

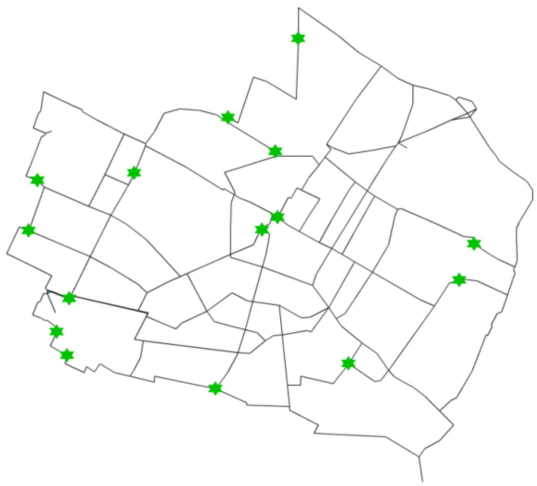

Figure 14: 15 Nodes minimizing TD in Modena

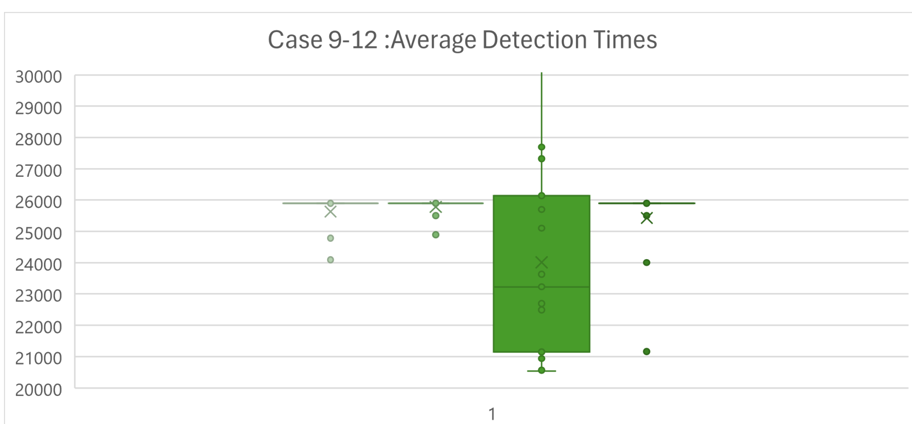

Figure 15: Avg. TD 9-12 (from left to right)

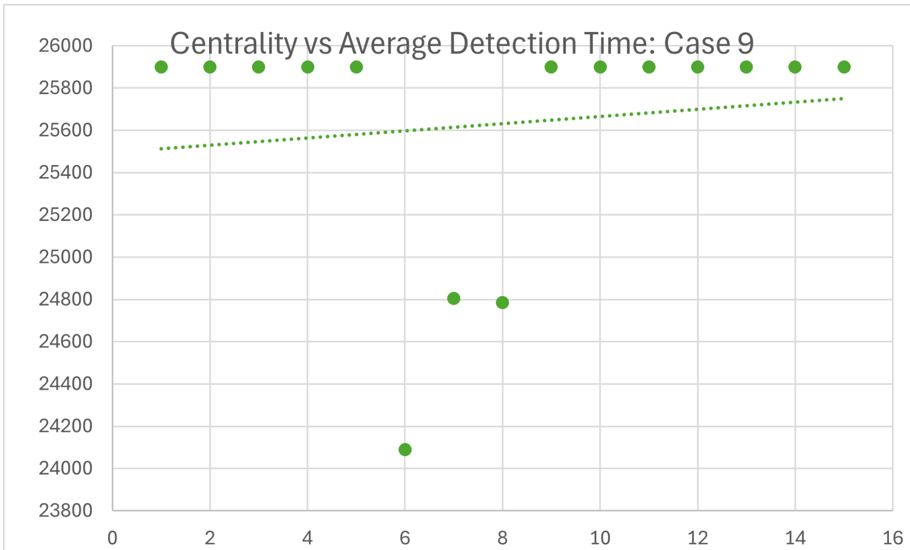

Figure 16: Central nodes vs Avg. TD

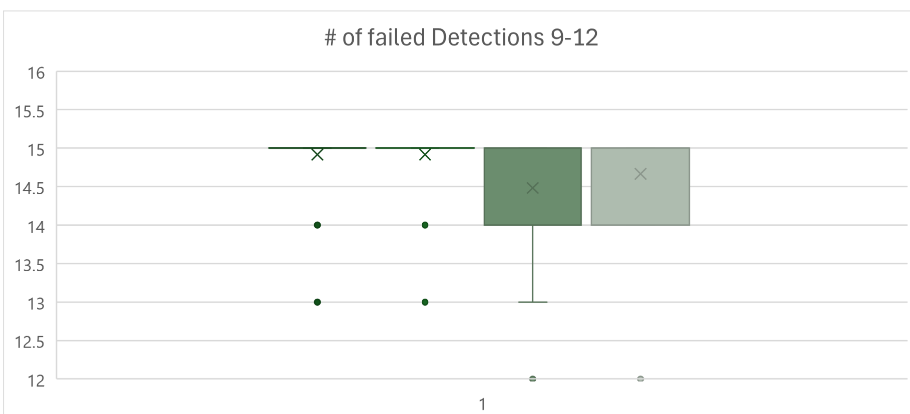

Figure 17: # of failed detections 9-12 from left to right

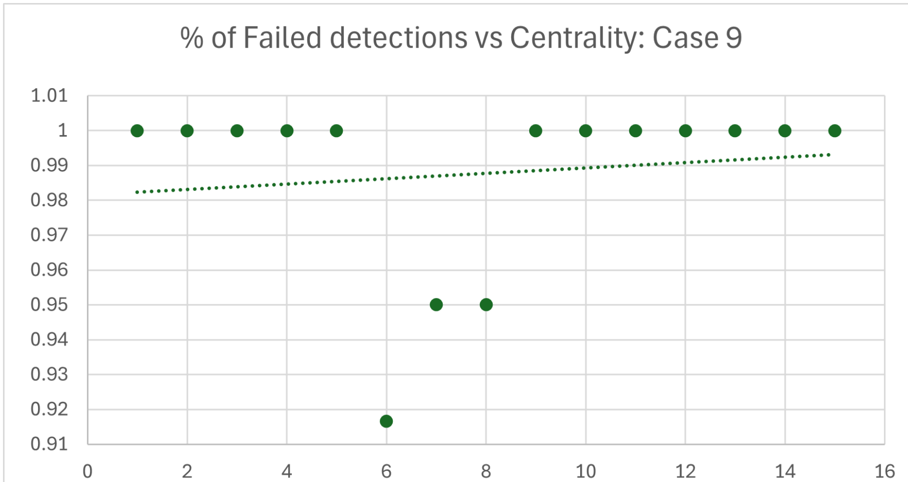

Figure 18: % of failed detections vs Centrality

Summary and ANOVA: Failed Detections

Anova: Single Factor Case1-4

SUMMARY

Groups	Count	Sum	Average	Variance
Case1	60	287	4.783333333	1.020056497
Case2	60	249	4.15	0.604237288
Case3	60	189	3.15	0.604237288
Case4	60	182	3.033333333	0.405649718

ANOVA

Source of Variati	SS	df	MS	F	P-value	F crit
Between Grou	125.8791667	3	41.95972222	63.71578195	3.26E-30	2.642851047
Within Group:	155.4166667	236	0.658545198			
Total	281.2958333	239				

Figure 19: ANOVA for Cases 1-4, failed detections

failed detections Case5-8

Anova: Single Factor

SUMMARY

Groups	Count	Sum	Average	Variance
Case5	60	552	9.2	17.58644068
Case6	60	552	9.2	17.58644068
Case7	60	472	7.866666667	18.15141243
Case8	60	419	6.983333333	15.71158192

ANOVA

Source of Variati	SS	df	MS	F	P-value	F crit
Between Grou	212.4458333	3	70.81527778	4.103100138	0.007300387	2.642851047
Within Group:	4073.116667	236	17.25896893			
Total	4285.5625	239				

Figure 20: ANOVA for Cases 5-8, failed detections

Failed detections						
Anova: Single Factor	Case9-12					
SUMMARY						
Groups	*Count*	*Sum*	*Average*	*Variance*		
Case9	60	895	14.91666667	0.111581921		
Case10	60	895	14.91666667	0.111581921		
Case11	60	869	14.48333333	0.728531073		
Case12	60	880	14.66666667	0.327683616		
ANOVA						
Source of Variati	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Grou	8.0125	3	2.670833333	8.350408479	2.67348E-05	2.642851047
Within Group:	75.48333333	236	0.319844633			
Total	83.49583333	239				

Figure 21: ANOVA for cases 9-12, failed detections

t-Test: Two-Sample Assuming Equal Variances

	Case 1	*Case2*
Mean	4.796610169	4.1694915
Variance	1.026884863	0.591467
Observatic	59	59
Pooled Var	0.809175921	
Hypothesiz	0	
df	116	
t Stat	3.786511195	
P(T<=t) one	0.000121795	
t Critical or	1.658095744	
P(T<=t) twc	0.000243591	
t Critical tw	1.980626002	

Figure 22: Two-tailed T-test between case 1 and 2, with p = 0.00024

t-Test: Two-Sample Assuming Equal Variances

	Case5	Case6
Mean	9.101694915	9.1016949
Variance	17.29982466	17.299825
Observatic	59	59
Pooled Var	17.29982466	
Hypothesiz	0	
df	116	
t Stat	0	
P(T<=t) one	0.5	
t Critical or	1.658095744	
P(T<=t) twc	1.980626002	
t Critical two-tail		

Figure 23: Two-tailed T-test between case 5 and 6

t-Test: Two-Sample Assuming Equal Variances		
	Case9	*Case10*
Mean	14.816667	14.916667
Variance	0.3217514	0.1115819
Observations	60	60
Pooled Variar	0.2166667	
Hypothesized	0	
df	118	
t Stat	-1.176697	
P(T<=t) one-ta	0.1208424	
t Critical one-	1.6578695	
P(T<=t) two-ta	0.2416848	
t Critical two-t	1.9802722	

Figure 24: Two-tailed T-test between case 9 and 10

11.2 NetworkX Code

```
# The following code is used to calculate the Betweenness Centrality for the networks
    import wntr
import networkx as nx
import matplotlib.pyplot as plt

# Load the water network from an .inp file
# inp_file = 'Net3.inp'
#network_names=['C:/Users/ahmed/Downloads/chama-main/chama-main/examples/Net3.inp','C:/Users

#inp_file = 'C:/Users/ahmed/Downloads/chama-main/chama-main/examples/Net3.inp'
# inp_file = 'C:/Users/ahmed/Downloads/chama-main/chama-main/networks/PA2.INP'
inp_file = 'C:/Users/ahmed/Downloads/chama-main/chama-main/networks/modena.inp'

# inp_file = 'C:/Users/ahmed/Downloads/chama-main/chama-main/networks/asce-tf-wdst/Anytown/A
# inp_file = network_names

wn = wntr.network.WaterNetworkModel(inp_file)

# Convert the water network model to a NetworkX graph
G = wn.get_graph()


# Calculate betweenness centrality
betweenness_centrality = nx.betweenness_centrality(G)

# Calculate node degree
node_degree = dict(G.degree())

# Get the top N nodes by betweenness centrality
# N_sensors = [1,3,5,10,15]
N_sensors = [15]

for n in range(0,len(N_sensors)):
    N = N_sensors[n]
    topNcentrality = sorted(betweenness_centrality, key=betweenness_centrality.get, reverse=
    print(topNcentrality)

    # Define the nodes to highlight
    nodes_to_highlight = topNcentrality

    # Extract node positions from the network model
    pos = {node: (wn.get_node(node).coordinates[0], wn.get_node(node).coordinates[1]) for no

    # Plot the network
    plt.figure(figsize=(24, 16))
```

```
# Draw nodes
nx.draw_networkx_nodes(G, pos, node_size=50, node_color='blue')

# Draw edges
nx.draw_networkx_edges(G, pos, alpha=0.5)

# Highlight specified nodes
nx.draw_networkx_nodes(G, pos, nodelist=nodes_to_highlight, node_size=20, node_color='re

# Add labels to the highlighted nodes
for node in nodes_to_highlight:
    plt.text(pos[node][0], pos[node][1], node, fontsize=12, color='black', ha='right')

plt.title('Net3 top 15 Central nodes',)
plt.axis('off')
plt.show()
```

11.3 MATLAB CODE

```
%%% THIS IS THE MATLAB FILE CONTAINING ALL THE FUNCTIONALITY THAT HAS BEEN
%%% USED IN MATLAB.
%%% It loads a .inp file, selects its nodes, generates a corresponding .msx
%%% file and simulates its hydraulic and biochemical interactions and
%%% stores the result to an array, for it to be further used.

clear; close('all'); clc;
start_toolkit;
% d = epanet('Net3.inp');
d = epanet('modena.inp');
% d = epanet('PA2.inp');
t_rials = 60
% node_count = d.getNodeCount
% initrandsensorArray = randi([1, node_count], 1, 15)
% sensor_index = initrandsensorArray
% sensor_index = [63, 62, 64, 65, 66, 61, 50, 49, 74, 48, 87, 75, 38, 57, 44] % NET3 INDICES
% sensor_index = [1, 12, 21, 28, 33 , 48, 49, 53, 60, 61, 63, 71, 78, 91, 41 ] % NET3 TEVA-S
% sensor_index = [103,124,142,202,197,196,194,179,174,97, 178,229,181,93,112] %PA2 INDICES
% sensor_index = [78,90,129,142,156,157,160,165,169,189,203,206,226,236,7] % PA2 TEVA-SPOT
% sensor_index = [41, 248, 40, 39, 38, 37, 193, 36, 194, 27, 26, 116, 247, 183, 49] % MODENA
sensor_index = [100,12,132,139, 142, 174, 195, 206 212, 217, 244, 263,27,38 79] % MODENA TEV
% sensor_index = [2, 3, 5];

results = ones(length(sensor_index),t_rials);
counter = 1
for trials = 1:t_rials
    disp(["trial #" num2str(counter)])
counter = counter+1
%% Filename

msx={};
msx.FILENAME = 'THMs.msx';
%% Section Title

msx.TITLE = 'Example: Net2tmp MSX TEST';
%% Section Options

msx.AREA_UNITS = 'FT2'; %AREA_UNITS FT2/M2/CM2
msx.RATE_UNITS = 'DAY'; %TIME_UNITS SEC/MIN/HR/DAY
msx.SOLVER = 'EUL'; %SOLVER EUL/RK5/ROS2
msx.COUPLING = 'NONE'; %COUPLING FULL/NONE
msx.COMPILER = 'NONE'; %COMPILER NONE/VC/GC
```

```
msx.TIMESTEP = 300; %TIMESTEP in seconds
msx.ATOL = 0.001;  %ATOL value
msx.RTOL = 0.001;  %RTOL value
%% Section Species
% <type> <specieID> <units> (<atol> <rtol>)

msx.SPECIES={
    'BULK    WaterAge    HR   0.01  0.001',
    'BULK    CL2    MG   0.01  0.001',
    'BULK    THMs        UG   0.01  0.001',
    'BULK    TOC         MG   0.01  0.001'}; %type [BULK/WALL] [specieID] [units UG/MG] [ato
%% Section Coefficients
%% CONSTANT name value % PARAMETER name value

msx.COEFFICIENTS = {
    'PARAMETER   Temperature    20',
'PARAMETER   Kw            0.3048 ',
'PARAMETER   Y             33.5',
'PARAMETER   Ka            0.02',
'PARAMETER   Kd           0.02 '}; %[name] [value]
%% Section Terms
% <termID> <expression>

msx.TERMS = {
    'Kf     (1.5826e-4 * RE^0.88 / D) ',
'Kb     (1.8e6*exp(-6050/(Temperature+273)))*TOC ;/day'}; % [termID] [expression]
%% Section Pipes
% EQUIL <specieID> <expression>
%
% RATE <specieID> <expression>
%
% FORMULA <specieID> <expression>

msx.PIPES = {'RATE        WaterAge        24',
'RATE        CL2        -Kb*CL2 - (4/D)*Kw*Kf/(Kw+Kf)*CL2 ',
'RATE        TOC             0 ',
'RATE        THMs            Y*Kb*CL2*1000'}; % [type] [specieID] [expression]
%% Section Tanks
% EQUIL <specieID> <expression>
%
% RATE <specieID> <expression>
%
% FORMULA <specieID> <expression>

msx.TANKS = {'RATE        WaterAge        24',
'RATE        CL2        -Kb*CL2',
```

```
'RATE        TOC             0 ',
'RATE        THMs            Y*Kb*CL2*1000'}; % [type] [specieID] [expression % [type] [spec
%% Section Sources
% <type> <nodeID> <specieID> <strength> (<patternID>)

msx.SOURCES = {'MASS 145 TOC 1000',
'MASS 111 TOC 1000',
'MASS 184 TOC 1000',
'MASS 217 TOC 1000'};

% List of node names
nodeNames = d.getNodeNameID;

% Number of quality conditions to generate
numConditions = 4;

% Species ID and values
speciesID = 'TOC';
values = [1000, 1000, 1000, 1000];

% Preallocate cell array for quality conditions
qualityConditions = cell(numConditions, 1);

% Generate random indices and create quality condition strings
for i = 1:numConditions
    randomIndex = randi(length(nodeNames));
    nodeName = nodeNames{randomIndex};
    qualityConditions{i} = sprintf('%s %s %s %.1f','MASS', nodeName, speciesID, values(i));
end

% Assign to msx.QUALITY
msx.SOURCES = qualityConditions;

% Display the generated quality conditions
disp('Generated Quality Conditions:');
disp(msx.SOURCES);%[CONC/MASS/FLOW/SETPOINT] [nodeID] [specieID] [strength] [patternID]
%% Section Quality Global
% GLOBAL <specieID> <value>

msx.GLOBAL = {'GLOBAL CL2 0.5'}; % [specieID] [value]
%% Section Quality
% NODE <nodeID> <bulkSpecieID> <value>
%
% LINK <linkID> <wallSpecieID> <value>
```

```
%% Section Parameters
% PIPE <pipeID> <paramID> <value>
%
% TANK <tankID> <paramID> <value>

msx.PARAMETERS = {''};
%% Section Patterns
% <patternID> <multiplier> <multiplier>

msx.PATTERNS = {''}; % [patternID] [multiplier]
%% Write MSX File.

msx
d.writeMSXFile(msx);
%% Load MSX File.

d.loadMSXFile(msx.FILENAME);
ComputedQualityNode = d.getMSXComputedQualityNode
d.unloadMSX;
%%  Compute Quality step by step.

d.solveCompleteHydraulics;
d.openQualityAnalysis;
d.initializeQualityAnalysis;
tleft=1; P=[];T=[];QsN=[]; QsL=[];
while (tleft>0)
    % Add code which changes something related to quality
    t = d.runQualityAnalysis;
    P = [P; d.getNodePressure];
    QsN = [QsN; d.getNodeActualQuality];
    QsL = [QsL; d.getLinkQuality];
    T = [T; t];
    tleft = d.stepQualityAnalysisTimeLeft;
end
d.closeQualityAnalysis;
%%  Load EPANET-MSX files.

d.loadMSXFile('Net2tmp.msx')
% List of node names
nodeNames = d.getNodeNameID;

% Number of quality ounteronditions to generate
numConditions = 4;

% Species ID and values
```

```
speciesID = 'THMs';
values = [0.1, 0.1, 0.1, 0.2];

% Preallocate cell array for quality conditions
qualityConditions = cell(numConditions, 1);

% Generate random indices and create quality condition strings
for i = 1:numConditions
    randomIndex = randi(length(nodeNames));
    nodeName = nodeNames{randomIndex};
    qualityConditions{i} = sprintf('%s %s %s %.1f','NODE', nodeName, speciesID, values(i));
end

% Assign to msx.QUALITY
msx.QUALITY = qualityConditions;

% Display the generated quality conditions
disp('Generated Quality Conditions:');
disp(msx.QUALITY);
%%  Compute Quality with MSX (specify type).

qual_res_MSX = d.getMSXComputedQualitySpecie('THMs')
%%  Get quality of specific nodes.

% sensor_index  = randi([1, d.getNodeCount], 1, 15)
sensors_names = d.getNodeNameID(sensor_index)
QN = d.getMSXComputedQualityNode(sensor_index)
%%
%%  Get species names.

type = d.getMSXSpeciesNameID;
disp(type)
%%  Get quality for specific species type (nodes and links).

species = type
MSX_comp = d.getMSXComputedQualitySpecie(species{3})

%%  Plot concentration for specific node and all species.

% d.plotMSXSpeciesNodeConcentration(1,1)
% d.plotMSXSpeciesNodeConcentration(1,2)
% d.plotMSXSpeciesNodeConcentration(1,3)
% d.plotMSXSpeciesNodeConcentration(1,4)


%  counter = 1
```

```
% for j = 1:4
% for i = 1:length(sensor_index)
%     % Get the current sensor index
%     x = sensor_index(i);
%
%     % Perform the action with the current sensor index
%     d.plotMSXSpeciesNodeConcentration(x, j);
%
%     name = num2str(counter)
%      saveas(gcf, name, 'png'); % Save as PNG file
%      counter = counter+1
% end
% end
% disp("amount of species ") ;disp(d.MSXSpeciesCount)
%
% results = ones(length(sensor_index),trials);
disp('Array')
disp(results);
intersectTimes(1,trials) = NaN;
% for j = 1:4

% Assuming 'd' is your EPANET MATLAB Toolkit object
speciesNodeConcentration = d.getMSXComputedQualitySpecie(type{3})
% Assuming speciesNodeConcentration contains the data in a format [time, concentration]

for i = 1:length(sensor_index)
x = sensor_index(i);
time = speciesNodeConcentration.Time ;
concentration = speciesNodeConcentration.NodeQuality(:,sensor_index(i));

% Value of y for the new line
newLineValue = 1; % Replace y with the actual value

% % Plot the original concentration vs time
% figure;
% plot(time, concentration, 'b');
% hold on;
%
% % Plot the new line
% plot(time, newLineValue * ones(size(time)), 'r--');
% legend('Concentration of THMs(UG/L)', 'Sensor Threshold Value');
% xlabel('Time(s)');
% ylabel('Concentration of THMs');
% title(['concentration of THMs at node', num2str(x)]);
```

```
% Interpolate the concentration data
interpConc = interp1(time, concentration, time);

% Find the intersection points
intersectIndices = find((interpConc(1:end-1) - newLineValue) .* (interpConc(2:end) - newLine
% intersectTimes = 0;
% intersectTimes(1,1) = 0;
% intersectTimes(2,1) = 2;
% intersectTimes(3,1) = 3;
% Calculate exact intersection times using linear interpolation
intersectTimes = time(intersectIndices) + (time(intersectIndices + 1) - time(intersectIndice
                 .* (newLineValue - interpConc(intersectIndices)) ./ (interpConc(intersectIn

% Display the intersection points
% disp('Intersection Points:');
% resultval(1,1) = 99;
try
    % intersectTimes(1,1);
    % results(i,1) = resultval(1,1);
    results(i,trials) = intersectTimes(1,1);
    % disp(intersectTimes);
    % disp(resultval);
catch
    % disp("nondetection DETECTED")
    results(i,trials) = NaN;
end

% % Mark the intersection points on the plot
% plot(intersectTimes, newLineValue * ones(size(intersectTimes)), 'ko', 'MarkerFaceColor', '
% hold off;

% end
end

end
```

11.4 EXCEL DATA

Summary and ANOVA: Failed Detections						
Anova: Single Factor		Case1-4				
SUMMARY						
Groups	Count	Sum	Average	Variance		
Case1	60	287	4.783333333	1.020056497		
Case2	60	249	4.15	0.604237288		
Case3	60	189	3.15	0.604237288		
Case4	60	182	3.033333333	0.405649718		
ANOVA						
Source of Variatio	SS	df	MS	F	P-value	F crit
Between Grou	125.8791667	3	41.95972222	63.71578195	3.26E-30	2.642851047
Within Group:	155.4166667	236	0.658545198			
Total	281.2958333	239				

Figure 25: ANOVA for Cases 1-4, failed detections

failed detections		Case5-8				
Anova: Single Factor						
SUMMARY						
Groups	Count	Sum	Average	Variance		
Case5	60	552	9.2	17.58644068		
Case6	60	552	9.2	17.58644068		
Case7	60	472	7.866666667	18.15141243		
Case8	60	419	6.983333333	15.71158192		
ANOVA						
Source of Variatio	SS	df	MS	F	P-value	F crit
Between Grou	212.4458333	3	70.81527778	4.103100138	0.007300387	2.642851047
Within Group:	4073.116667	236	17.25896893			
Total	4285.5625	239				

Figure 26: ANOVA for Cases 5-8, failed detections

Failed detections						
Anova: Single Factor		Case9-12				
SUMMARY						
Groups	Count	Sum	Average	Variance		
Case9	60	895	14.91666667	0.111581921		
Case10	60	895	14.91666667	0.111581921		
Case11	60	869	14.48333333	0.728531073		
Case12	60	880	14.66666667	0.327683616		
ANOVA						
Source of Variatio	SS	df	MS	F	P-value	F crit
Between Grou	8.0125	3	2.670833333	8.350408479	2.67348E-05	2.642851047
Within Group:	75.48333333	236	0.319844633			
Total	83.49583333	239				

Figure 27: ANOVA for cases 9-12, failed detections

t-Test: Two-Sample Assuming Equal Variances

	Case 1	*Case2*
Mean	4.796610169	4.1694915
Variance	1.026884863	0.591467
Observatic	59	59
Pooled Var	0.809175921	
Hypothesiz	0	
df	116	
t Stat	3.786511195	
P(T<=t) one	0.000121795	
t Critical or	1.658095744	
P(T<=t) twc	0.000243591	
t Critical tw	1.980626002	

Figure 28: Two-tailed T-test between case 1 and 2, with p = 0.00024

t-Test: Two-Sample Assuming Equal Variances		
	Case5	*Case6*
Mean	9.101694915	9.1016949
Variance	17.29982466	17.299825
Observatic	59	59
Pooled Var	17.29982466	
Hypothesiz	0	
df	116	
t Stat	0	
P(T<=t) one	0.5	
t Critical or	1.658095744	
P(T<=t) twc	1.980626002	
t Critical two-tail		

Figure 29: Two-tailed T-test between case 5 and 6

t-Test: Two-Sample Assuming Equal Variances		
	Case9	*Case10*
Mean	14.816667	14.916667
Variance	0.3217514	0.1115819
Observations	60	60
Pooled Variar	0.2166667	
Hypothesized	0	
df	118	
t Stat	-1.176697	
P(T<=t) one-ta	0.1208424	
t Critical one-t	1.6578695	
P(T<=t) two-ta	0.2416848	
t Critical two-t	1.9802722	

Figure 30: Two-tailed T-test between case 9 and 10