Universiteit
Leiden

Master Computer Science

Centrality-based NSGA-II method for optimal sensor placement in water distribution networks

Name:	Tao Peng
Student ID:	s3076326
Date:	23/01/2023
Specialisation:	Computer Science: Data Science
1st supervisor:	Dr. Michael Emmerich
2nd supervisor:	Dr. Iryna Yevseyeva
3rd supervisor:	Dr. Kegong Diao

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

Monitoring the water distribution network to maintain the security of the water supply is meaningful and challenging work. Placing sensors in the network to detect the contamination event is a common and effective way to achieve the goal. It is a multi-objective optimization problem (MOP) to decide where to install sensors. Many studies have been conducted on developing approaches for MOP; however, scant attention has been paid to centrality metrics. Moreover, the centrality-based approach simply used nodes, leading to limited search space and diversity. In this study, we propose an approach to combining the two approaches, aiming to explore a large enough decision space and benefit from the centrality metric. To this end, we select two objectives as our optimization objectives, namely the minimum detection and the detection likelihood (coverage). We use a preselected subset of nodes as candidates to reduce the computation load. NSGA-II is used as the backbone optimization algorithm; in addition, we analyze the distributions of centrality and use them as heuristic information to design the mutation operator. We test the algorithm on the benchmark network, BWSN1. After experiments, Pareto fronts and hypervolume indicators are used to analyze the performance results, and we found that combined centrality methods show better performance.

Acknowledgements

I want to start by expressing my gratitude to my supervisors, Dr. Michael Emmerich, Dr. Kegong Diao, and Dr. Iryna Yevseyeva, for their considerate guidance. When I get into challenging circumstances, I can quickly get in touch with them and receive helpful advice. In addition to this, they teach me the fundamentals of scientific research. For me, they unlocked a gateway to the academic universe.

I also want to express my gratitude to my beloved wife for her unending support; without her, I could not accomplish anything.

Contents

1 Introduction 1
- 1.1 General motivation and explanation of the topic 1
- 1.2 Related work 2
- 1.3 Research Question 2
- 1.4 Approach 2
- 1.5 Structure of the thesis 3

2 Problem Definition 4
- 2.1 Water distribution network 4
- 2.2 Multiobjectives optimization problem 4
- 2.3 NSGA-II 7

3 Methods 11
- 3.1 Centrality Metrics Discussion 11
- 3.2 Preselection 12
- 3.3 Centrality-based mutation in NSGA-II 13
- 3.4 2-dimensional analysis 14
- 3.5 Evaluation 15
- 3.6 Experimental setup 15

4 Results 18
- 4.1 Centrality distribution and Interpretation 18
- 4.2 Multiple Pareto Fronts 18
- 4.3 Hypervolume indicator 20
- 4.4 Interpretation of Results 20

5 Conclusions **22**
5.1 Summary 22
5.2 Response to the research question 22
5.3 Future work 23

References **24**

Chapter 1

Introduction

1.1 General motivation and explanation of the topic

In modern society and cities, the water supply system is a fundamental infrastructure. As the scale of towns expanded, the water supply system became more complex, which also increased the risk to water security. There are different kinds of threats, such as water leaking, water pollution, or other disasters occurring in the water supply systems. In this study, we focus on water contamination.

Contamination means an unexpected substance is dissolved in water. This is harmful to public health, especially when the contamination substance is toxic. The contamination substance could be easily spread out through the connected pipes in the water supply system, increasing the influence further. To make the water supply system robust, deploying an efficient and accurate contamination detection solution in a water distribution system is necessary. A commonly used method is placing sensors at specific locations in the water supply system. Once a contamination event occurs, when the contaminated substance carried by the water flow passes by one of the sensors, it could be detected, and if the concentration exceeds a threshold, the event could be sent to the relevant departments, and they would take action, such as locating and repairing.

There are several aspects to the efficiency of a sensor detection solution, such as the detection time, the population affected prior to detection[17], accuracy, the number of sensors used, and so on. Different sensor placement solutions would lead to performance differences in these aspects. However, determining optimal positions in a network for sensor placement is a typical NP-hard problem. The problem is hard because the number of possible positions to place the sensors grows exponentially with the number of sensors.

1.2 Related work

The Battle of Water Sensor Networks (BWSN)([17]) design challenge was conducted to compare various approaches to solving this problem for the water distribution network. There are four designed objectives: the expected detection time, the expected population affected, the expected amount of contaminated water consumed, and the detection likelihood. Since then, many studies have been conducted on this Multiple-objective Optimization Problem(MOP), and different approaches are implemented, such as genetic algorithm(GA) based([6], [19], [13]), greedy search based([2], [15], [10]), heuristics based([1], [11]), and Particle swarm algorithm based([12], [16]). However, scant attention has been paid to the network's exclusive metrics from the aspect of network/graph theory, such as centrality.

Besides, in some studies that are centrality-based ([18], [21], [20]), they simply use the nodes with high centrality as sensor placement candidates, which leads to the search space being restricted and the diversity being damaged.

Therefore, in this study, we proposed an approach to combine the two approaches, aiming to explore a large enough decision space and benefit from the centrality metric.

1.3 Research Question

As discussed above, for this MOP, we decide to use an evolutionary algorithm (EA) integrated with centrality to fulfill the task. Then, the main research question we have is:

- How centrality metric could be integrated within an EA?
- Is the centrality metric beneficial for the sensor placement problem?

1.4 Approach

Firstly, we will conduct a preselection over all the nodes in the water distribution network. This would be helpful to reduce computation workload, and different preselections will be discussed and compared.

Then, we will analyze and formulate the sensor placement problem in mathematical form. We will use NSGA-II([4]) as our backbone algorithm, and based on the framework of the algorithm, we will convert the problem into chromosomes, individuals, and a population set. A centrality-based mutation operator will be tested.

The experiments are divided into three parts: preselection, simulation, and optimization. From the preselection, we get the candidate set of all nodes for further optimization. During the simulation period, we will construct all the possible contamination events and run simulations on a hydrodynamic simulator. By doing this, we can get a detection time matrix recording the minimal detection time for each node for all the possible contamination events. We can use this matrix to compute the optimization objective functions. Finally, we run the optimization algorithm with different sensor number settings.

1.5 Structure of the thesis

The remainder of this thesis is structured as follows: In Chapter 2, we will discuss the problem we will solve in more detail, including the properties of the water distribution network that we will study, the concepts of MOP, and the sensor placement problem formulation in the MOP context. The standard version of NSGA-II will be explained as well.

In Chapter 3, we will introduce the methods we used, and different centrality metrics are explained, including not only their definition but also the internal meaning they would stand for, and according to this, we propose our way to utilize the centralities; After that, we compare three preselection methods and determine the candidate set we will work on. Then, we will introduce the centrality-based mutation method we designed for NSGA-II.

We show and discuss the results in Chapter 4, including the centrality distribution, Pareto fronts, and hypervolume changes.

Finally, we have a conclusion in Chapter 5, we give a summary of this thesis and list the possible future work.

Chapter 2

Problem Definition

In this chapter, we will introduce the problem that we work on, explain the background mathematical concepts, and formulate the problem as a mathematical problem. The standard version of NSGA-II will then be explained in more detail.

2.1 Water distribution network

A water distribution network is an infrastructure to carry water from waterworks or wells to satisfy different kinds of consumers. It is composed of many different kinds of components, such as pipes, junctions, pumps, valves, and tanks. Typically, in a water distribution network, pipes are denoted as edges, and junctions or tanks are denoted as nodes. In this paper, we focus on the Battle of the Water Sensors Network 1 (BWSN1)([17]). This network consists of 126 junctions, 168 pipes, one constant head source, two pumps, two tanks, and eight valves. The layout of the network is shown in Figure 2.1.

2.2 Multiobjectives optimization problem

There are classes of problems for which, when we try to solve them, we have to make trade-offs between two or more conflicting objectives in order to arrive at a solution, in other words, multiple criteria decision-making. For example, when we construct a building, we want to minimize the investment cost and maximize the durability at the same time. A formal mathematical definition is given as follow [7]:

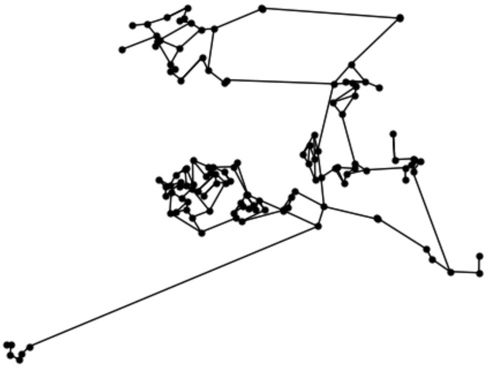

Figure 2.1: Layout of Battle of the water sensors network1(BWSN1)

Definition 2.2.1 (Multiobjective Optimization)**.** Given m ($m \geq 2$) objective functions $f_1 \colon \mathcal{X} \to \mathbb{R}$, ..., $f_m \colon \mathcal{X} \to \mathbb{R}$ which map a decision space $\mathcal{X}$ into $\mathbb{R}$, a multiobjective optimization problem (MOP) is given by the following problem statement:

$$\text{minimize}\, f_1(x), \ldots, \text{minimize}\, f_m(x), x \in \mathcal{X}$$

When we try to apply optimization to these problems, more than one objective function should be optimized. However, there is no single solution that can optimize each objective simultaneously, which means if one objective improves, then at least one of the other objectives will get worse. To compare the two solutions, the definition of Pareto dominance is given as follows:

Definition 2.2.2 (Pareto Dominance)**.** A solution x^1 is said to Pareto dominate a solution x^2, if and only if: $\forall i \in \{1, \ldots, n\} : f_i(x^1) \leq f_i(x^2)$ and $\exists i \in \{1, \ldots, n\} : f_i(x^1) < f_i(x^2)$.

where f_i is the i-th objective function. With this definition, the objective sets could be arranged in Pareto order. The definitions of Efficient Point, Efficient Set, and Pareto Front are as follows:

Definition 2.2.3 (Efficient Point, Efficient Set, Pareto Front)**.** [7] The minimal elements of the Pareto order on decision space are called efficient points.

The subset of all efficient points in decision space is called the efficient set.
The subset of all non-dominated objective points in objective space is called the Pareto Front.

There are four quantitive design objectives proposed in [17], in this project, we choose two of them to optimize, namely the expected time of detection and the detection likelihood.

Expected time of detection, for a particular contamination event, the time of detection by a sensor is the elapsed time from the start of the event to the first time detecting a nonzero contamination concentration. In a particular sensor placement solution with n sensors, t_j refers to j-th sensor's detection time. The time of detection for this sensor placement solution, t_d, is the minimum among all the sensors in the solution.

$$t_d = \min_j t_j, j \in \{1, \ldots, n\}$$

The objective function to be minimized is the average detection over all possible contamination scenarios,

$$F_1 = \frac{1}{m} \sum_{i=1}^{m} t_d \rightarrow min$$

Where m is the total number of all contamination scenarios.

Detection likelihood is an indicator for the probability of detection of a contamination event, it is estimated by

$$F_2 = \sum_{i=1}^{m} d_r$$

where d_r=1 if contamination event r is detected, otherwise zero. m is the total number of possible contamination events. To convert the maximization to minimization, we use the inverse of F_2 as the optimization objective

$$F_2 = \frac{1}{\sum_{i=1}^{m} d_r}$$

In addition, the sensor budget could serve as another objective, since in a practical environment we prefer to find solutions that have good performance with fewer sensors. However, to be consistent with the settings in BWSN, we chose to set the sensor budget as the constraint. Besides, we will try more sensor budget settings to observe the impact of sensor budgets, and the result will be shown later.

To simulate contamination events occurring in the network, each node will be injected with a certain amount of contaminated substance, and each injection is a single contamination event, so in the BWSN1, the total number of possible contamination events is 129 (126 junctions, 1 head source, and 2 tanks). The injection starts at the beginning of the simulation and lasts for two hours; the total simulation duration is 96 hours.

In a solution to the problem of sensor placement, every node has only two choices: be chosen for placement or not, so this is a binary optimization problem. When we design the chromosome, it is straightforward to use binary encoding. In each chromosome sequence, each bit represents a node, if that node is chosen for sensor placement, the corresponding bit is assigned with 1, otherwise 0.

When we try to install sensors in the water distribution network, the sensor budget is a problem that must be considered. In the BWSN challenge, different sensor budget options are given; we will adopt these and try more options.

After the previous analysis, the problem we study could be formulated as follows:

$$
\begin{aligned}
F_1 &= \frac{1}{m}\sum_{i=1}^{m} t_d \rightarrow min \\
F_2 &= \frac{1}{\sum_{i=1}^{S} d_r} \rightarrow min \\
& s.t. \\
\sum x_i &\le n \\
x_i &\in \{0,1\}
\end{aligned}
$$

where x_i denotes the bits in the chromosome, n is the sensor budget.

2.3 NSGA-II

An evolutionary algorithm (EA) is a generic population-based optimization method inspired by biological evolution. Similar to the process of evolution, there are four typical components:

- **reproduction**: generates new offspring from the population
- **recombination**: gene crossover from parents' chromosome
- **mutation**: gene mutation occurred at some points of chromosome
- **selection**: select parents from the population set

For a problem to be optimized, a candidate solution is considered an individual in the whole population set. After every generation, the better offspring will be retained, which is called the "elitist strategy". After generations of evolution, the result will converge on a local or global optima.

The algorithm we used in this study, NSGA-II([3]), is a Pareto dominance-based multiobjective evolutionary algorithm (MOEA). The main loop of NSGA-II is given by Algorithm 1.

Firstly, the population set is initialized, it can be randomly generated under the constraints. Then, to produce one new individual of the next generation, three steps should be executed, namely *select_parents*, *crossover*, and *mutate*. In the process of *select_parents*, two parents are selected from the population set. There are different kinds of selection methods; we choose the tournament selection method, which selects two tournaments from the population set, and the winner of each tournament is used for crossover. The comparison between two individuals is based on a ranking method that will be explained later.

The next step is the crossover for the parents we got in the previous step. We recombine the chromosomes to generate new ones, and we choose the single-point crossover method shown in Figure 2.2. In this method, one crossover point is randomly selected, and the segment between the beginning of the chromosome and the crossover point is copied from one parent, and the rest is copied from the other parent. With this operation, we can get two new chromosomes. The crossover operator makes sure that features in the current population set can be passed down from one generation to the next; otherwise, the process of evolution would be totally random.

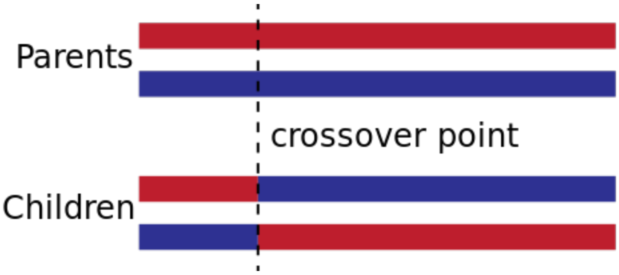

Figure 2.2: Single point crossover, the top red and blue bars denote the chromosomes of parents and the bottom two bars represent the recombined children's chromosomes.

Then, the following step is mutation. Upon the chromosomes generated in the previous step, some bits in the chromosomes are selected and inverted, randomly selecting the mutation position is a simple and common way. In this paper, we proposed a centrality-based method to identify the mutation position, and this will be explained later in more detail. The mutated chromosomes are considered the offspring, and they are added to the offspring set. The mutation step is beneficial for producing new features in the offspring, which means the diversity of the gene pool increases.

All the steps described before can be considered the variation part; in this part, new individuals are generated, and after combining them with the current population set, we get a double-sized population set. To keep the better half of the combined population set as the next generation, a two-level ranking method is applied over the population set.

The first-level ranking is non-dominated sorting; the algorithm is given in Algorithm 2. After the non-dominated sorting process, the population set is divided into partitions with different ranks; the individuals in the partitions with higher ranks are better than the ones in the partitions with a lower rank. Inside each partition, the crowding distance is computed for every individual. A crowding distance is the accumulated distance from the neighbor individual along each objective. It is described in Algorithm 3, and it reflects the crowding degree, an individual who has a high crowding distance indicates it is located in a sparse position and is more different from others. In NSGA-II, it is encouraged to choose individuals with a higher crowding distance for crossover and the next generation to maintain diversity.

Algorithm 1 NSGA-II

```
Initialize population $P_0 \subset \mathcal{X}_\mu$
while not terminate do
    $Qt \leftarrow \emptyset$
    for all $i \in \{1, \ldots, \mu\}$ do
        $(x_1, x_2) \leftarrow select\_parents(P_t)$
        $r^i \leftarrow crossover(x_1, x_2)$
        $q_t^i \leftarrow mutate(r^i)$
        $Q_t \leftarrow Q_t \cup q_t^i$
    end for
    $(R_1, \ldots, R_l) \leftarrow non\_dom\_sort(P_t \cup Q_t)$
    $P_{t+1} \leftarrow$ Find the top $\mu$ best out of $P_t \cup Q_t$
    $t \leftarrow t + 1$
end while
```

Algorithm 2 non-dominated sorting

```
Population $P$
Rank $k \leftarrow 0$
while $p \neq$ do
    $k \leftarrow k + 1$
    $R_k$ $\leftarrow$non-dominated solutions in $P$
    $P \leftarrow P \setminus R_k$
end while
return $R_1, \dots, R_k$
```

Algorithm 3 crowding-distance sorting

```
$R$, partitions with the same non-dominance rank
$r = |R|$
for $j$ = 1,..., m (number of objectives) do
    $L \leftarrow$ sort $R$ by $j$-th objective ascendingly
    for $i$ = 1, ..., —$R$— do
        $c_i = 0$
        $x \leftarrow$ $i$-th individual in $R$
        $m, n \leftarrow$ upper and lower neighbor to $x$ in $L$
        $c_i += (m_j + n_j)$, $m_j, n_j$ are the $j$-th objectives of $m, n$
    end for
end for
return crowding distance $c_1, \dots, c_r$
```

Chapter 3

Methods

In this chapter, we will introduce the method we used in more detail. First, we will discuss the centrality metrics from different angles. Then, preselection methods will be explained. Also, we will give a description of the centrality-based mutation method. The following are the evaluation methods and experimental settings.

3.1 Centrality Metrics Discussion

For element-level analysis, centrality represents the importance of a node in a network, from different aspects. The simplest measure of centrality is degree centrality([9]), which is the number of edges connected to the node.

Betweenness centrality([8]) is another centrality measure computed based on the shortest path in the network. It is computed by the equation:

$$g(v) = \sum_{s \neq t \neq v} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

where $\sigma_{st}(v)$ represents the number of shortest paths between vertex s and t that passes through v, σ_{st} is the total number of the shortest path between vertex s and t. Betweenness is used to measure the degree to which a vertex plays the role of a "bridge" in a network or graph. When moving to a water distribution network, a node with high betweenness centrality means it connects more parts of the network together, so if a sensor is placed at this position, the detection probability would be higher than at other nodes.

Eigenvector centrality is used to measure the influence of a node within a network. The concept is that a node's influence is also determined by its neighbors' importance. Connections to high-scoring nodes contribute more to the score of the node in question than equal connections to low-scoring nodes. It is computed by the equation:

$$e(v) = \frac{1}{\lambda} \sum_{i \in V} \alpha_{v,i} x_i$$

In this equation, λ is the eigenvalue of the adjacency matrix, $\alpha_{v,i}$ is 1 when the two vertex v and i are connected, otherwise 0. x_i is the eigenvector centrality of vertex i, it could be initialized to 1. After iteratively computing across the network, the final result will converge, and then we get the eigenvector centrality of each vertex. In the water distribution network, a node with higher eigenvector centrality means it could be an important facility, such as a reservoir or waterworks, so it is a good choice for sensor placement.

To retain the features of both betweenness centrality and eigenvector centrality, we proposed a method called combined centrality, which is a weighted sum of these two. As aforementioned, every kind of centrality represents importance in a single aspect. However, for a real-world network, for example, a water distribution network, we make decisions only relying on one single centrality, which would be insufficient and, more than this, would lead to biased results. In a centrality combined with more than one centrality, different centralities contribute to the final centrality. The following equation is used to compute the combined centrality

$$c = 0.5 * b(v) + 0.5 * e(v)$$

where 0.5 is the weight for each centrality, which means the two centralities have the same contribution to the final centrality. Other weight possibilities could be investigated in further research.

3.2 Preselection

The water distribution network used in this paper consists of a total of 129 nodes. The total number of possible solutions with x sensors would be the x-combination of 129 nodes, namely $\binom{129}{x}$, which will be extremely huge. To reduce the computation load, it is necessary to apply a pre-processing method to gain a subset of total nodes; at the same time, the result generated from this subset should not be worse than that from all nodes. There two ways are considered.

a), Diao et al. [5] developed a method based on controllability analysis to pre-select nodes as sensor placement candidates, which are called driver nodes. A minimum of 47 nodes is enough for full control of contaminant intrusions.

b), Another way is based on centrality. As aforementioned, the nodes with higher centrality are more important, so it is straightforward to treat these nodes as candidates. Based on these approaches, we get three groups of candidates, as shown in Table 3.1. After experiments based on these candidates, the hypervolume indicator result is shown in Figure 3.1. We can see that the subset based on controllability has the best performance, so this subset will be used to provide candidates in the following experiments.

Table 3.1: Three preselection candidate sets, generated by controllability, betweenness centrality, and eigenvector centrality

approach	candidates nodes
controllability	7, 8, 10, 13, 14, 16, 21, 26, 36, 37, 38, 39, 41, 42, 45, 48, 50, 52, 64, 66, 72, 73, 74, 76, 80, 82, 83, 84, 85, 91, 93, 99, 100, 101, 106, 110, 112, 113, 114, 118, 123, 124, 125, 126, 129, 130, 131
betweenness centrality	1, 20, 21, 22, 23, 30, 31, 34, 37, 47, 48, 49, 54, 58, 59, 60, 61, 62, 63, 64, 65, 68, 69, 70, 72, 73, 75, 76, 77, 78, 79, 80, 81, 82, 85, 86, 87, 88, 89, 90, 91, 92, 97, 98, 99, 109, 110
eigenvector centrality	33, 50, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101

3.3 Centrality-based mutation in NSGA-II

In this section, we will introduce the centrality-based mutation improvement used in NSGA-II. The original approach is random mutation, which means the mutation position is randomly chosen from the chromosomes, and each position has the same probability of being chosen. However, as aforementioned, in a water distribution network, each node has a different centrality or importance, and this will lead to different contributions to the detection time and coverage. It is straightforward that we hope the nodes with higher centrality have more chances to be investigated. The method is given in Algorithm 4. As described in the algorithm, the nodes with the top 10 highest centrality values will be assigned 5 times higher weights than other nodes,

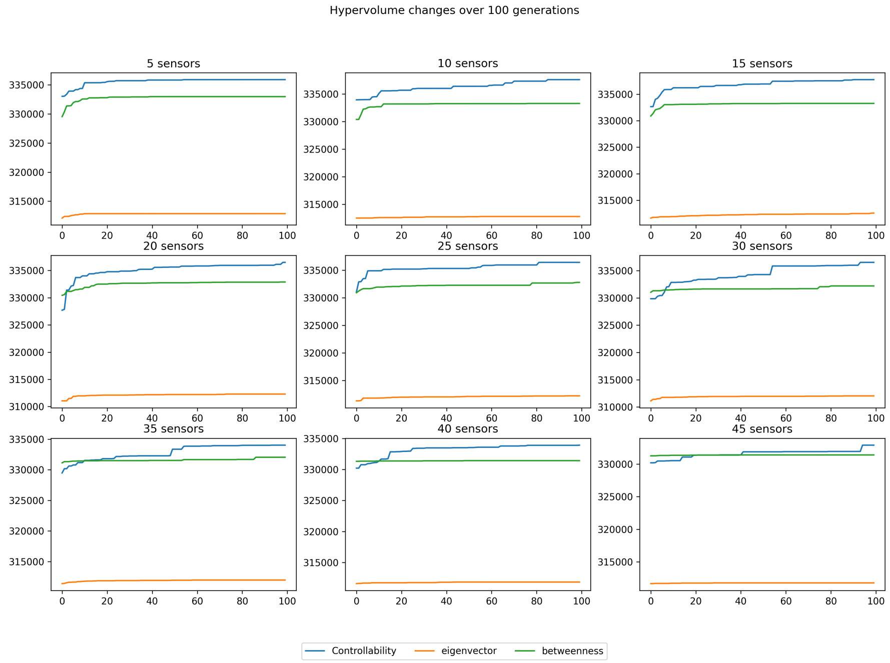

Figure 3.1: Hypervolume changes with different sensor budget settings, generated from experiments with three different preselection candidate sets.

which means by this way we try to make sure that the nodes with higher centralities have a higher probability to be selected for mutation.

As discussed in Section 3.1, we have three centrality options, namely betweenness centrality, eigenvector centrality, and combined centrality. Besides, we will use the original version mutation as a baseline. The experiments will be based on these four mutation approaches.

3.4 2-dimensional analysis

As discussed in Section 2.2, we have two objective functions to optimize, so we can use a 2D Pareto front to visualize the results of previous algorithms. A typical example of a 2D Pareto front is as shown in Figure 3.2. In this figure, the two axes represent the two objectives, respectively; the scatter line is the nondominated set (Pareto front), and the dominance direction is top-right. The left part of the Pareto front is the dominant space, which is the space to seek improvement. The shape of the Pareto

Algorithm 4 Choose mutation position

```
$S \leftarrow$ preselected candidate nodes are sorted by the centrality descendingly
$upper = 10 * 5 + (|S| - 10)$
$rand \leftarrow$ randomly draw a integer from $[1, upper]$
$c = 5$
for $i \in \{1, \ldots, |S|\}$ do
    if $rand \leq$c then
        return $i$
    else
        if $i \leq 10$ then
            $i = i + 5$
        else
            $i = i + 1$
        end if
    end if
end for
```

front could be varied, but all the points on the Pareto front are nondominated by each other. In this paper, we use the horizontal axis for coverage and the vertical axis for detection time.

3.5 Evaluation

We use the hypervolume indicator to evaluate the performance of results generated from experiments. The hypervolume indicator measures the size of the dominated space, bound from above by a reference point [7], as shown in Figure 3.3, where r is the reference point and the size of the gray area is the hypervolume. For the fixed reference point, the greater the hypervolume indicator, the bigger the size of the dominated space, which means better performance. In this study, we choose the upper bound value of both objectives as the reference point, avoiding a negative hypervolume indicator. The simulation duration is 96 hours (96 * 3600 seconds), and the maximum uncoverage indicator is 1, so the reference point is [345600, 1].

3.6 Experimental setup

This section contains a description of the experimental setup. The software and tools we used are listed as follows:

- Python 3.9, the environment for the experiment.

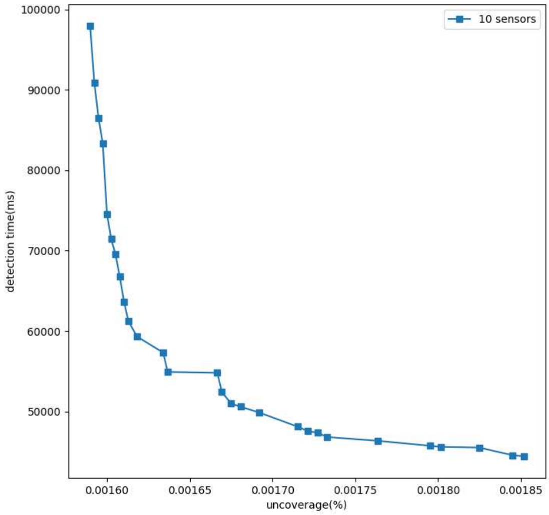

Figure 3.2: An example of 2D Pareto front, the two axes represent two objectives.

- Wntr[14], a Python package to simulate the resilience of water distribution networks.
- EPANET 2.0, software used to model a water distribution network.
- Networkx, a Python package to compute the centrality metrics.
- Matplotlib, a Python pack to visualize the results.
- NSGA-II, implemented in the Python programming language.

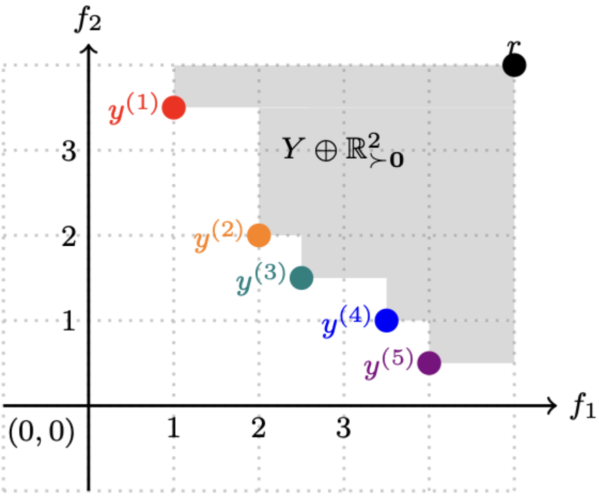

Figure 3.3: Hypervolume indicator computation example, $y^{(i)}$ is the nondominated point on the Pareto front, r is the reference point, and the size of the gray area is the hypervolume dominated by the Pareto front. [7]

Chapter 4

Results

4.1 Centrality distribution and Interpretation

We first investigate the distribution of betweenness centrality, eigenvector centrality, and combined centrality. Figure 4.1 shows the distribution of centralities across all nodes in the network. We can see that higher centrality is concentrated in the ranges junction-20 to junction-35 and junction-55 to junction-101, at some junctions, they have only one nonzero centrality, and after combination, they would be paid more attention to. In a word, a single centrality focuses on a smaller range than the combined centrality.

Table 4.1 shows the correlation matrix between betweenness centrality and eigenvector centrality, and we can see that there is a weakly positive correlation between them.

Table 4.1: The correlation matrix between betweenness centrality and eigenvector centrality.

	betweenness	eigenvector
betweenness	1.000000	0.312631
eigenvector	0.312631	1.000000

4.2 Multiple Pareto Fronts

As aforementioned in Section 3.3, we have four mutation approaches, and for each approach, we test a different number of sensors, namely $5, 10, 15, 20, 25, 30, 35, 40$, and 45.

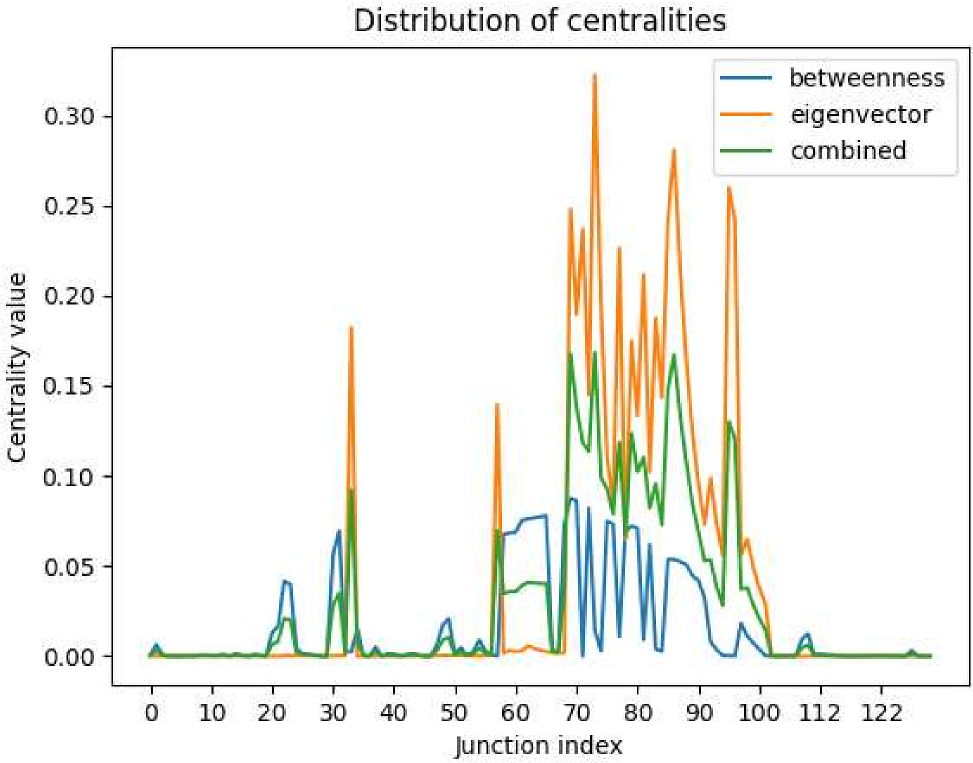

Figure 4.1: The distributions of different centralities. The horizontal axis is the junction index and the vertical axis is the centrality value.

Figure 4.2 shows the Pareto fronts result after simulations. The four images represent different mutation operators, and in each image, the horizontal axis is the uncoverage and the vertical axis is the detection time counted in seconds. Different colored dots represent the Pareto front with different sensor numbers. The same pattern is that, as the number of sensors increased, both objectives improved. However, after the number of sensors reaches 30, the improvement is little. This is consistent with the discussion in Section 3.2.

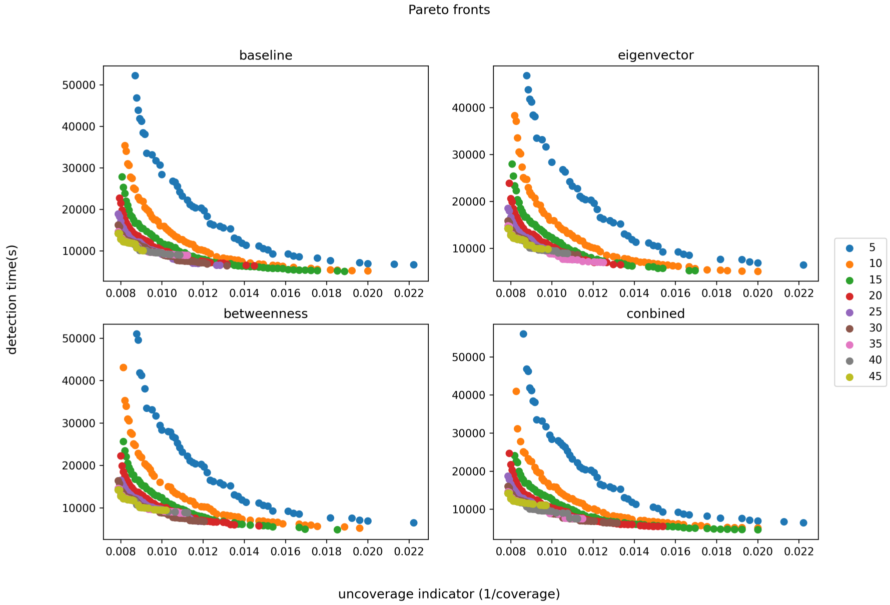

Figure 4.2: Pareto fronts for various sensor number settings. The horizontal axis is the uncoverage indicator, and the vertical axis is the minimum detection time. The top-left image is the Pareto fronts for baseline mutation, the top-right image is for eigenvector centrality-based mutation, the bottom-left image is for betweenness centrality, and the bottom-right image is for combined centrality. Different color dots represent the Pareto fronts for different sensor budgets.

4.3 Hypervolume indicator

Figure 4.3 shows the hypervolume changes under different mutation operators and sensor numbers. In each image, the horizontal axis is the generation number, and the evolution repeats for 100 generations; the vertical axis is the hypervolume indicator.

4.4 Interpretation of Results

From the results above, all four mutation approaches based on NSGA-II can converge, as can be seen both from Figure 4.2 and Figure 4.3. It indicates that NSGA-II is suitable for solving such problems. Specifically, in the top-left image, when the sensor number is 5, the maximum uncoverage (worst case) indicator is around 0.022, which means the detected scenario number is about 12(1/0.022), and the maximum detection time is around 50,000 seconds. After the sensor number increases to 30, the ideal

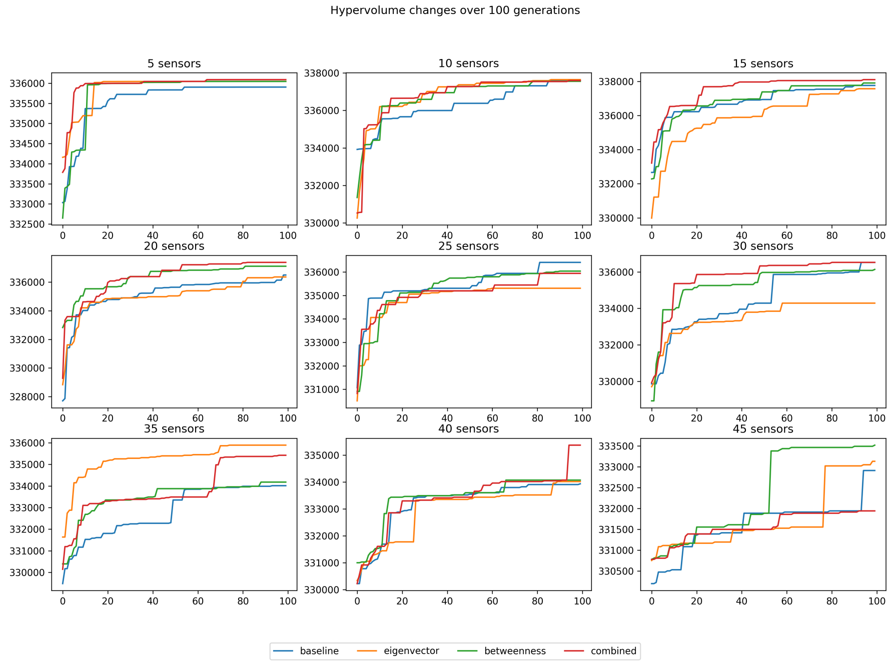

Figure 4.3: Hypervolume changes for various sensor number settings. The horizontal axis is the generation number, and the vertical axis is the hypervolume. Each image is for one sensor number. In each image, blue, yellow, green, and red lines denote baseline, eigenvector centrality, betweenness centrality, and combined centrality, respectively.

number as discussed in Section 4.2, the maximum uncoverage drops to 0.013, which means it detects around 77 scenarios out of 129 and the maximum detection time is about 15000 seconds.

In Figure4.3, the hypervolume indicator reveals the performance difference between these four mutation approaches. For the final converged hypervolume (greater is better), we can see that the combined centrality, which we proposed in this study, shows better performance in 6 settings out of 9 (5, 10, 15, 20, 30, 40), the betweenness centrality has the second-best performance, taking the second position in 4 settings (5, 15, 20, 25, 40), and the first position in 1 setting (45). When talking about the convergence speed, we cannot see any obvious pattern.

Chapter 5

Conclusions

5.1 Summary

In this study, we investigated the probability of solving a sensor placement problem using the multiple objectives optimization approach. Starting from this thought, we tried to use the standard version algorithm, NSGA-II, an EA-based algorithm, to solve this problem on a benchmark network. We also proposed the idea of using the centrality metric of the network to improve the performance of the algorithm, namely, centrality-based mutation. Different preselection methods are compared to reduce the computation workload. After experiments with different sensor numbers and mutation methods, we find that NSGA-II is suitable to solve the sensor placement problem. When increasing the number of sensors placed in the network, both the detection time and coverage improve significantly. However, after the number reaches a threshold, there is less room for improvement. What's more, the centrality-based mutation we proposed, specifically the combined centrality mutation, shows the best performance compared to other approaches.

5.2 Response to the research question

When coming back to the question we proposed at the beginning of this thesis, for the first question, to integrate the centrality metric within the NSGA-II, we designed a centrality-based mutation operator, by this approach, the mutation is expected to occur more likely on the nodes with higher centrality.

To answer the second question, we found that simply using the nodes with high centrality as preselection is not beneficial for the performance, only taking one centrality into account for centrality-based mutation is also not helpful. A form of weighted summation of two kinds of centrality shows a performance improvement.

5.3 Future work

We conduct experiments on a network with 129 nodes, which is a small-scale network; in the future, we will test the algorithm with large-scale networks, such as BWSN Network 2, a network with 12,523 nodes.

Centrality is a big family; there are many other centralities we can investigate, such as closeness or pagerank centrality. In addition, trying to find out the practical meaning of centrality in the context of the water distribution network is helpful to understand centrality better.

Finally, in this study, we proposed the combined centrality method. In the experiments, we simply set the same weight for two kinds of centrality. We think that it is not the only choice; different weight settings could be explored, especially when we combine three or more types of centrality.

References

[1] Mustafa M Aral, Jiabao Guan, Morris L Maslia, et al. “Optimal design of sensor placement in water distribution networks”. In: *Journal of Water Resources Planning and Management* 136.1 (2010), p. 5.

[2] Jonathan Berry et al. “Sensor placement in municipal water networks with temporal integer programming models”. In: *Journal of water resources planning and management* 132.4 (2006), pp. 218–224.

[3] K. Deb et al. “A fast and elitist multiobjective genetic algorithm: NSGA-II”. In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197. DOI: 10.1109/4235.996017.

[4] Kalyanmoy Deb et al. “A fast and elitist multiobjective genetic algorithm: NSGA-II”. In: *IEEE transactions on evolutionary computation* 6.2 (2002), pp. 182–197.

[5] Kegong Diao and Wolfgang Rauch. “Controllability analysis as a pre-selection method for sensor placement in water distribution systems”. In: *Water Research* 47.16 (2013), pp. 6097–6108. ISSN: 0043-1354. DOI: https://doi.org/10.1016/j.watres.2013.07.026. URL: https://www.sciencedirect.com/science/article/pii/S0043135413006003.

[6] Demetrios Eliades and Marios Polycarpou. “Iterative deepening of Pareto solutions in water sensor networks”. In: *Water Distribution Systems Analysis Symposium 2006*. 2008, pp. 1–19.

[7] Michael T. M. Emmerich and André H. Deutz. “A tutorial on multiobjective optimization: fundamentals and evolutionary methods”. In: *Natural Computing* 17.3 (Sept. 2018), pp. 585–609. ISSN: 1572-9796. DOI: 10.1007/s11047-018-9685-y. URL: https://doi.org/10.1007/s11047-018-9685-y.

[8] Linton Freeman. “A Set of Measures of Centrality Based on Betweenness”. In: *Sociometry* 40 (Mar. 1977), pp. 35–41. DOI: 10.2307/3033543.

[9] Linton C. Freeman. “Centrality in social networks conceptual clarification”. In: *Social Networks* 1.3 (1978), pp. 215–239. ISSN: 0378-8733. DOI: https://doi.org/10.1016/0378-8733(78)90021-7. URL: https://www.sciencedirect.com/science/article/pii/0378873378900217.

[10] Dinesh Kumar Gautam, Prakash Kotecha, and Senthilmurugan Subbiah. “Efficient k-means clustering and greedy selection-based reduction of nodal search space for optimization of sensor placement in the water distribution networks.” In: *Water Research* (2022), p. 118666.

[11] Jiabao Guan et al. “Optimization model and algorithms for design of water sensor placement in water distribution systems”. In: *Water Distribution Systems Analysis Symposium 2006*. 2008, pp. 1–16.

[12] Cheng-yu Hu et al. “Sensors placement in water distribution systems based on co-evolutionary optimization algorithm”. In: *2015 1st International Conference on Industrial Networks and Intelligent Systems (INISCom)*. IEEE. 2015, pp. 7–11.

[13] Chengyu Hu et al. “Modified NSGA-III for sensor placement in water distribution system”. In: *Information Sciences* 509 (2020), pp. 488–500.

[14] 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 & Software* 95 (2017), pp. 420–431. ISSN: 1364-8152. DOI: https://doi.org/10.1016/j.envsoft.2017.06.022. URL: https://www.sciencedirect.com/science/article/pii/S1364815216309501.

[15] Andreas Krause et al. “Efficient sensor placement optimization for securing large water distribution networks”. In: *Journal of Water Resources Planning and Management* 134.6 (2008), pp. 516–526.

[16] Malvin S Marlim and Doosun Kang. “Optimal water quality sensor placement by accounting for possible contamination events in water distribution networks”. In: *Water* 13.15 (2021), p. 1999.

[17] 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 (Nov. 2008), pp. 556–568. DOI: 10.1061/(ASCE)0733-9496(2008)134:6(556).

[18] Robert Paluch et al. “Optimizing sensors placement in complex networks for localization of hidden signal source: A review”. In: *Future Generation Computer Systems* 112 (2020), pp. 1070–1092.

[19] Ami Preis and Avi Ostfeld. “Multiobjective sensor design for water distribution systems security”. In: *Water Distribution Systems Analysis Symposium 2006.* 2008, pp. 1–17.

[20] Do Guen Yoo et al. “Applications of network analysis and multi-objective genetic algorithm for selecting optimal water quality sensor locations in water distribution networks”. In: *KSCE Journal of Civil Engineering* 19.7 (2015), pp. 2333–2344.

[21] Xizhe Zhang et al. “Identification of efficient observers for locating spreading source in complex networks”. In: *Physica A: Statistical Mechanics and its Applications* 442 (2016), pp. 100–109.