OPTIMAL SENSOR LOCATIONS IN WATER DISTRIBUTON NETWORKS

by
GÜLER KIZILENİŞ

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University
July 2006

OPTIMAL SENSOR LOCATIONS IN WATER DISTRIBUTION NETWORKS

APPROVED BY:

Assist. Prof Kemal Kılıç _____
(Thesis Advisor)

Assoc. Prof Ş. İlker Birbil _____

Assist. Prof Gürdal Ertek _____

Assist. Prof Kerem Bülbül _____

Assist. Prof Nihat Kasap _____

DATE OF APPROVAL:

to my family

Acknowledgements

I would gratefully like to thank my supervisor Kemal Kılıç due to his encouragement, guidance, suggestions, and understanding throughout the development of this thesis and his patience.

I cannot fully express my gratitude, love, and thanks to Cumhur, İlknur and Çağlayan Kızıleniş "my family" for their morale support, encouragement, patience and endless love throughout my education and life.

My special thanks goes to Cenk Aydın for his great help, morale support during my graduate studies.

And finally, I would also like to thank to my friends Canan, Bahar, Volkan and Ali for their friendship.

OPTIMAL SENSOR LOCATIONS IN WATER DISTRIBUTON NETWORKS

Güler KIZILENİŞ

Industrial Engineering, MS Thesis, 2006

Thesis Supervisor: Asst. Prof. Kemal KILIÇ

Keywords: Optimal Sensor Locations, Expected Time to Detect, Expected Population Exposed, Simulated Annealing, Tabu Search, and Clustering

Abstract

Following the attacks of September 11,2001 in the United States, concerns over terrorist attacks have sharpened. The attacks highlighted that public water distribution systems are inherently vulnerable to accidental or intentional water contamination because of their geographical distribution, relative isolation and possible impact to public health. There is an ongoing and fertile research environment aiming to prevent such threats in water distribution systems. One of the key problems arising in the field is to identify sensor locations that minimize the impacts (e.g. detection time, population exposure, contaminated volume consumed) of a terrorist attack aiming the water distribution systems.

In this thesis, we present a model of the sensor placement problem in municipal water networks. In our formulation, we focus to the sensor configuration that minimizes the expected time to detection and the expected population exposed. The first objective ensures that all such attacks should be detected in a very short time. We formulate these problems as binary nonlinear objectives, and propose a solution methodology framework based on meta-heuristics, namely, simulated annealing and tabu search. Two different neighborhoods generation mechanisms are utilized in the heuristics. We simulated the contaminant transport in EPANET in order to derive flow and velocity information. A novel approach is introduced in order to handle different flow patterns occurring during the day. In order to ensure the full detection probability, a new concept of restricted sensors that is located at the isolated node in the network is commenced. In the proposed framework, the close relation between clustering and location problems is exploited in the sense that the initial solutions of the meta-heuristics are generated with a k-medoids clustering algorithm.

İÇMESUYU DAĞITIM AĞLARINDA SENSÖRLERİN OPTİMAL YERLEŞTİRİLMESİ

Güler KIZILENİŞ

Endüstri Mühendisliği, Yüksek Lisans Tezi, 2006

Tez Danışmanı: Yar. Doç. Kemal KILIÇ

Keywords: Optimal Sensör Yerleri, Beklenen Tespit Zamanı, Beklenen Maruz Kalan Nüfusu, Yapay Isıl İşlem, Tabu Araştırma ve Sınıflama

Özet

11 Eylül 2001, Amerika Birleşik Devletleri saldırısına takiben, terörist saldırılarına olan ilgi arttı. Saldırılar, coğrafi dağılımdan, göreceli tenhalıktan ve halk sağlığına mümkün etkisinden dolayı, kazara ya da kasıtlı olan su kirliliği, halk suyu dağıtım sistemlerinin savunmasızlığını öne çıkardı. Su dağıtım sistemlerinde ki bu gibi tehditleri engellemeyi amaçlayan, gelişmekte olan ve zengin bir araştırma konusu vardır. Bu alandaki anahtar problemlerden biri, su dağıtım sistemlerindeki saldırılarının etkilerini (örn: tespit zamanı, maruz kalan nüfus, tüketilen kirli su hacmi) asgariye indirmeyi amaçlayan, sensör yerlerinin belirlenmesidir.

Bu tezde, içme suyu ağlarındaki sensör yerleştirme problem modelini sunduk. Formülasyonumuzda, beklenen tespit zamanı ve beklenen maruz kalan nüfusu asgariye indiren sensör konfigürasyonuna odaklandık. İlk hedef, tüm bu gibi saldırıları iyi bir çözümle bulmayı sağlamalıdır. Bu problemleri, ikili düzlemsel olmayan hedefler olarak ve yapay zeka optimizasyon algoritmaları yani, yapay ısıl işlem ve tabu araştırma algoritmalarına dayalı, çözüm yolunu kesin ve açık olarak anlattık. Yapay zeka optimizasyonunda kullanılan iki değişik komşu yaratma mekanizması kullandık. Akış ve hız verilerini türetmek için EPANET'den kirliliğin taşınımına benzetim yaptık. Gün boyunca meydana gelen değişik akış modelleriyle başa çıkmak için yeni bir yöntem tanıtıldı. Tam tespiti sağlamak için, şebekedeki ölü noktalara yerleştiren kısıtlı sensörler fikri ortaya konuldu. Önerilen çözümde, sınıflama ve yerleştirme problemleri arasında öyle bir yakın ilişki ortaya koyduk ki yapak zekâ problemi için olan başlangıç çözüm k-medoids sınıflandırma yöntemiyle yaratıldı.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Water distribution systems are organized for delivering water to consumers. Under normal operating conditions, complex networks of pumps, pipes, and storage tanks provide high quality treated water to consumers. Water distribution systems are composed of sources, pipes and hydraulic control elements such as; valves, tanks, and reservoirs. Figure 1.1 depicts an overview of a generic water distribution system. Fresh water is pulled from a reservoir, treated and stored in a series of holding tanks and pumping stations. Water is then distributed via a network of underground pipes [9].

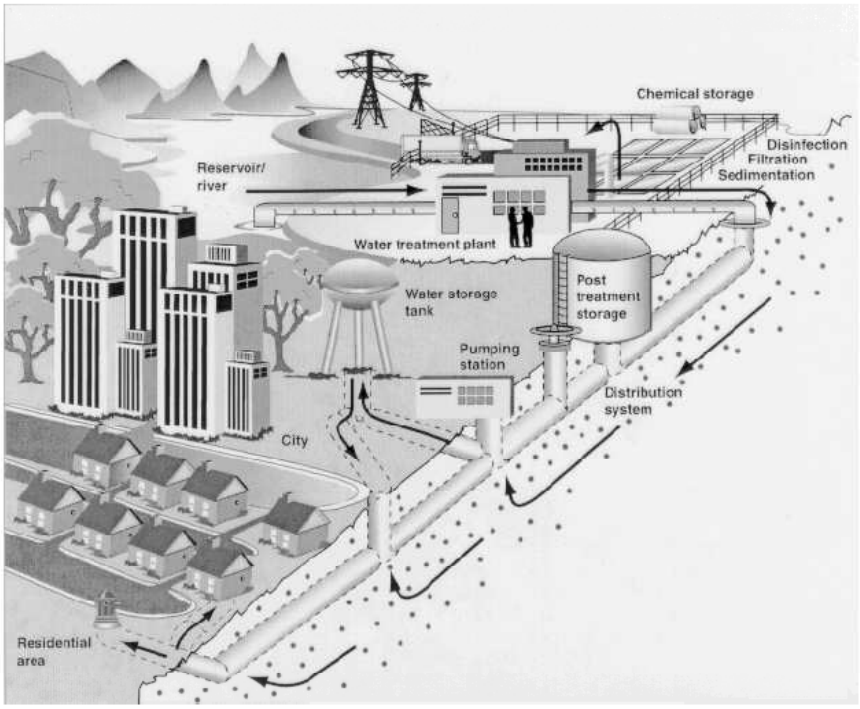


Figure 1.1 Representative Drinking Water Distribution System [9]

A distribution system of pipelines, pipes, pumps, storage tanks, and the appurtenances such as various types of valves, meters, and so on offers the greatest opportunity for terrorism because it is expensive, relatively unprotected, accessible and often isolated.

There are four types of threats in water distribution systems; cyber threats such as attacks on central control system to create simultaneous failures, physical threats such as destroying part of the network, chemical threats such as introducing nerve agents, blister agents, cyanides, arsenic, mercury, and so on, and biological threats such as putting in pathogens.

Water distribution systems are spatially varied and many of the system components such as tanks and pumps are located in isolated locations. Water distribution systems, therefore, have a natural potential to be vulnerable to a variety of threats- physical, chemical, and biological- that may jeopardize the systems reliability of delivering safe water. Particularly, the raw water source (surface or groundwater), raw water channels and pipelines, raw water reservoirs, treatment facilities, connections to the distribution system, pump stations, valves and finished water tanks and reservoirs are vulnerable to terrorist activities.

The ability of a water distribution system to provide water to its customers can be cooperated by destroying or disrupting key physical elements of the water system. The most important elements include raw water facilities (dams, reservoirs, pipes, and channels), treatment facilities, and distribution system elements (transmission lines and pump stations). Physical disruption may result in significant economic cost, problem, and loss of confidence by customers, but has limited direct threat to human health.

Contamination is usually pointed out as the most serious potential terrorist threat to water systems. Chemical or biological agents could extend throughout a distribution system and result in sickness or death among the consumers. It is not feasible to identify all

possible contaminants but it can be classified into various categories such as; biological weapons, chemical weapons, biotoxins, viruses, parasites, pesticides, toxic chemicals. The attack scenarios are unpredictable and patterns of water supply and consumer demand change every day, which further complicates the prevention of such attacks [19].

The need to predict the distribution of pollutants in water systems and to monitor their concentrations at various system locations is major concerns in order to maintain the safety of water supplied to the public. Tracking pollutant movement and concentrations within a water distribution system is not an easy task. It requires a mathematical quantity and quality model for conducting an accurate extended period hydraulic water quantity and quality simulation and the ability to monitor real-time pollutants' concentrations.

Various water quality and early warning detection systems models are proposed in the literature. The aim of water quality models is to maintain water quality in a desired level. Water quality in a distribution system deteriorates basically because of two main reasons; a decay or growth of nonconservative components that takes place during the transport process and an accidental or targeted intrusion of pollutants as we discussed above. The behavior of nonconservative components can be predicted using dynamic water quality simulation models like EPANET [28]. There are several other models also available for modeling both the hydraulics and water quality in drinking water distribution system. However, most effective and widely used one is the EPANET, which is developed by U.S. Environmental Protection Agency. EPANET is an extended period hydraulic water quality and quantity simulation model, and simulates the fate of the water, as well as the possible contaminants, within the water distribution system, given the structural parameters such as the pipe sizes, pump locations, and so on, the demand conditions at different time periods and the parameters of the contaminant intrusion. The EPANET provides an output of the hydraulic simulation. The output includes the time history of the flow velocities for each pipe in the network in order to calculate contamination timing in the network for each possible number of attack scenarios.

In case of external intrusion of contaminants, monitoring stations should be designed to provide alarm signal within acceptable time limit. Early Warning Systems (EWS) concept is introduced to avoid and immediately mitigate the impacts of a contaminant such as death, illnesses among the consumers in the water distribution system. An effective EWS should consider the following; exhibits warning in sufficient time for action, provides affordable cost, requires low skills and trainings, covers all the potential threats, identifies the source, allows remote operation [12].

There are four key elements in EWS: 1) Alarm level is needed to identify the concentrations at which the agents pose a threat to human health so that alarms can be activated. The response started by an alarm is a notice to public to stop using the water. 2) Fate and transport of pathogens and chemicals. Chemicals and microbiological agents can act in a variety of ways as they migrate through a water system. In water distribution systems the control of pathogens and chemicals is typically accomplished by incorporating disinfection agents, at the supply source. Free chlorine is one of the most widely used disinfectants. Free chlorine levels are usually maintained between minimum and maximum level. A target agent is known to chemically degrade at a certain rate in the presence of free chlorine. It may be possible to use a hydraulic/chemical model of the distribution system, like EPANET, to determine the fate and transport of the agent concentration through the system.3) Sensor Locations; the location and density of sensors in a warning system is impacted by the result of the system characterization, vulnerability assessment and threat analysis. Placement of sensors in distribution system must be suitable because it is one of the most vulnerable points in a water system. However, even if sensors can be optimally located within a distribution system, there would be probably not enough time to prevent exposure of a part of the public to contaminated water. Monitoring conducted within the distribution system provides time to limit exposure, isolate the contaminated water, and initiate mitigation/remediation actions 4) Responding to a contamination threat once a contamination event has been recognized and described. The determination of the fate of the contaminant in the system is crucial for an effective response. This requires both locating various sensors at different positions in the system and understanding of the

hydraulic behavior of the water in the distribution system. The last step is to wash out the contaminant from the system and confirm that the contaminant is no longer present. Therefore an EWS is a necessity for designing a secure water distribution system particularly in post-9/11 era [19].

To sum up, there has been increased interest in development of sensor networks to detect accidental and deliberate contamination events in water distribution systems and so facilitate corrective action or notification. Optimization models and algorithms have been developed using various metrics to locate a limited number of sensors that minimize the impact of such contamination events. These optimization models are usually based on simplifying assumptions about design objectives, network contaminant transport, sensor response, event detection, emergency response, and installation and maintenance costs. Therefore, their real life performances are unknown. Furthermore, the problem can be computationally complex to solve for large networks.

In this thesis we are aiming to propose a methodology, which might be utilized in order to determine sensor locations in the water distribution systems. We consider two objective functions; namely, the expected time to detection and expected population exposed. We propose two techniques based on local search heuristics; namely, simulating annealing and tabu search, and test their performances in various experimental conditions. We also test the effectiveness of determination of the initial sensor locations with clustering. Other than the solution methodology, the contributions of this study are: 1) introducing a restricted sensor concept in a way that all contaminant intrusion is detected and 2) simulating the flows, which are obtained from EPANET on the time dimensions more realistically than the existing techniques 3) presenting the new methodology for the sensor placement problem in water distribution networks.

In the following part, we introduce a detailed literature review and discuss various optimization problems related with the efficient water distribution systems. Solution techniques for locating on-line contamination monitoring stations, sensor placement in distribution water networks are discussed. Next, problem definitions, the assumptions of

the model and methodology are detailed and coupled with a methodology pinpointing the major issues that should be considered while designing the system and during the implementation phase of the study. In the same section, we introduce the proposed algorithms for the solution of the problem. Next part is dedicated for the experimental results and the discussion of the major findings. We conclude the thesis with our final remarks and future works.

2 LITERATURE REVIEW

The analysis of water quality in water distribution network systems is relatively new. In 1990, Environmental Protection Agency (EPA) introduced new regulations, which require water quality standards must be met at the consumer taps rather than at the source treatment plants. This initiated the need for water quality modeling and raised other problems and research needs.

More recently, concerns over terrorist attacks have been sharpened following the September 11,2001 attacks in the United States. Terrorist activities are potentially tragic, and the existence of such threats affects public confidence in water supplies. So 9-11 attacks highlighted that public water distribution systems are inherently vulnerable to accidental or intentional water contamination because of their geography.

Both the change in water quality regulations and concerns over terrorist activities catalyzed the research on the topic. In addition to other problems, the optimal placement of the sensors draws relatively high attention. The current literature can be categorized into two groups: 1) Location of on-line water quality monitoring stations 2) Sensor placement in municipal water networks. In the existing literature, studies related with location of on-line water quality monitoring stations were undertaken before the *security era* and monitor the quality of the water. The water quality in distribution systems deteriorates because of two reasons: The first one is self-decay or growth of water quality constituents, the other one is an external pollution intrusion. Locations of on-line water quality monitoring stations concepts started up because of the need for dealing with water quality decrease with time and distance from the source. For this problem, the aim is to keep water quality within acceptable limits.

When the focus of interest is external pollution, in which detection time is most critical, in the case of terrorist activities, the second category of sensor placement in municipal water networks are discussed in the literature. A number of algorithms have been developed to solve the problem of placing a limited number of sensors in a water distribution system so that health protection from accidental or targeted contaminant intrusions is maximized. Our problem shows the similarity with the concept of sensor placement in municipal water networks. Since, in the case of external contamination intrusion, we have dealt with how locating the sensors throughout the water distribution networks minimize the impacts of such an attack, now, we will discuss the existing literature on these two problems.

First problem is related to the location of on-line contamination monitoring stations. Efficient water quality monitoring is one of the most vital tools to guarantee a reliable water supply to consumers of drinking water distribution systems. A decrease in pressure at one or more system nodes can reduce quantities supplied to consumers, while the accidental entry of contaminants or self-deterioration of the water quality within the network itself can damage public health [8]. The control of pressure is usually attained by booster pumps and pressure reducing valves, while water quality can be controlled by booster chlorine injections [6, 24, 31], and by monitoring. The problem of booster chlorine injections is to add a minimum amount of disinfectant at different system locations at different times, while retaining a minimum threshold disinfectant concentration level at each of the consumers' nodes.

Because the injection process does not affect water flow, Boccelli et al. [6] showed that if chlorine decay follows a first order reaction, then the response at a node follows a linear reaction corresponds to an injection at booster location.

Here, the choice becomes the location of boosters and their injection rates. Boccelli et al. [31] directed their attention to the problem of locating disinfectant booster stations that minimize the dosage required to maintain residuals throughout the supply network, and

introduced a useful parameterization for disinfectant source types by building on the previous work. The same problem is formulated and explained as a location problem and is solved by a mixed integer linear programming problem by Pool [25].

In the literature, EPA regulations require that samples be taken at locations represents the water quality in the systems. Because of this reason, several approaches were developed to define the *Representative Concept*.

In addition, Lee et al. [18] presented an integer programming formulation for the location problem of on-line monitoring stations in order to maximize the quality of the representativeness water samples. In their methodology the best set of stations is proposed to be one that maximizes the total demand coverage. The demand coverage is defined as the percentage of the demand monitored by a set of monitoring stations. Furthermore, with some degree of belief, one can also infer the water quality of the upstream nodes that feed the node where monitoring station is located. According to Lee et al., an upstream node (i.e. coverage of an upstream node) can be measured as the fraction of the water monitored at the downstream node, where the station is located and water is passed through it. Lee et al. solved that problem as an integer programming method. Next, Kumar et al. [15] developed an algorithm for the same problem based on a heuristic which eliminates the use of the computationally cumbersome integer programming. But in this model, no proof of global optimality was shown. In contrast, Al-Zahrani, et al. [2] proposed a genetic algorithm to solve the problem, which is the identification of the water quality monitoring stations in a water distribution system.

From an engineering point of view, the demand coverage method [2, 15, 18] has some restrictions. First, it only considers steady-state water quality conditions that are usually not accomplished. Next, the method does not consider the residence time that water spends in the system and sequential variation of the water quality. Lastly, information is only considered in the upstream direction and coverage does not extend in the downstream direction.

In order to solve these limitations, Harmant et al. [11] adapted the objective function to introduce time dependence and water quality into the demand coverage model in order to recognize nodes with lower water quality.

Moreover, Woo et al. [34] further modified the same objective function by applying weights at each term by normalizing the concentrations by the source values. Similar to the previous research, only the water quantity was considered as a factor of monitoring importance. Even if the water quality as well as the water quantity was considered, the retention times in the system were used to estimate the water quality at the network nodes. Thus nodes with lower quality received higher weights in the objective function.

All of the mentioned models claim that if the downstream water quality is acceptable then the water supplied before reaching that node should be acceptable. So, the solution methodology involves an installation of monitoring stations at downstream locations. This is the same idea for the sensor placement models except installing monitors at a node will delay or miss the detection intrusions until many consumers have been affected by the contaminant.

A variety of approaches have been developed for sensor-placement problems in water networks. Firstly, Ostfeld et al. [21] introduced a service level constraint for detecting random accidental contaminant intrusions. They defined the level of service provided to the consumers before the maximum volume of consumed polluted water prior to detection. Their methodology involves the establishment of an auxiliary network that represents all possible flow directions for a typical demand cycle and finding the shortest paths between the nodes, a pollution matrix to represent the domains of detection and coverage for each node and a *set covering* algorithm to optimally allocate the monitoring stations. The nodes of the auxiliary network are the original nodes and directed edges are placed between two nodes if there is a direct flow (i.e. pipe) between them. The length of the edges is equal to the average travel time between the nodes. The flows between the nodes and the velocity of the flow during a day under different demand conditions are determined by the EPANET.

The calculation of flow pattern has several limitations in this paper [21]. The representative flow pattern is calculated according to formula that replaces the flow of one pipe by flow of two parallel pipes. Then the pipes are represented on the network by two undirectional and parallel pipes. Note that different demand patterns occur at different time points in a day. Residential areas usually consume more water in the evenings; on the other hand business neighborhoods demand more water during the daytime. These different demand patterns yield different flows during 24-hours. The difference can be so drastic that at different time points even the direction of the flow in a pipe changes depending on the time point. Ostfeld's et al. approach averages the flows that occur in a pipe throughout 24-hours. However this approach yields misleading conclusions. For example, if a pipe from node i to node j has a flow from node i to node j, between 8 AM and 4 PM, but the direction of the flow changes afterwards, Ostfeld's et al. approach assumes that there is a certain (average) flow at both directions throughout the day. Based on this assumption a contaminant introduced from node i can be determined at node j (given that a sensor is located at node j) within an acceptable time determined by the average flow. On the other hand, in reality if the flow direction changes after 4 PM, an intrusion cannot be detected until the direction of the flow changes. Therefore, the calculations based on the averaging technique of Ostfeld et al. are misleading and should be corrected. In this study, the calculation of flow patterns showed diversity from the Ostfeld's et al. proposed flow scenario. The used flow pattern is detailed in Chapter 3.

Next, Kumar et al. [16] examined Ostfeld et al. [21] paper in a discussion note. They argue to use a *time-based service level* constraint rather then the *volume-based service level* constraint as proposed by the authors. Ostfeld et al. disagree with Kumar's et al. suggestion that the contamination node determination algorithm, but it referred to the fact that the calculations were based on averages, hence the study would not represent reality, therefore, would yield inaccurate results. Note that this discussion note is one of the first references mentioning the contaminant node determination problem.

Ostfeld et al. extended their previous work [21] through explicitly considering the deterioration and dilution of water quality as distributed with flow, and by taking into account the monitoring equipment capabilities to detect pollutant concentrations [22]. In their previous paper, the water dilution and water quality decay properties of the contaminant were not taken into consideration. In the follow up paper, a minimum hazard level (MHL) was introduced and a node was considered contaminated only if the concentration of the contamination was over the MHL. The proposed methodology is the same as the methodology proposed in their earlier paper [22].

All of the aforementioned models assume single injections of contaminant. But, Ostfeld et al. [23] included multiple random injections to the previous proposed model. Also they constructed the pollution matrix by genetic algorithm, which is a different construction method than given in the previous one [21].

Laird et al. [17] also used the multiple injection term. But the difference from the Ostfeld et al. [23] is that they introduced unknown, time dependent injection terms at every node in the network and formulated this as a quadratic program to solve for the time profiles of the injections. Ostfeld's injections term introduced not at every node in the network, also the sources of injections are assumed known. The approach was able to identify unique solutions and find good solutions with a minimum number of injection locations. Until, the source is determined, many consumers may die until that time because of not considering the detection time.

Berry et al. [3] presented an optimal sensor configuration that minimizes the expected fraction of the population at risk. The likelihood of a contamination is modeled as a fixed probability distribution across junctions in the network, which can be used to model the likelihood of either accidental or intentional attacks. This paper differs from the others via sensor placement, since it is assumed that sensors are placed at only edges of the network. Also the service level aspect is taken into the objective function and the sensors

are added to the constraints. However, the extended simulation hydraulic equations were used in this study, Berry et al. derived only the flow direction from EPANET.

On the contrary, Uber et al. [32] showed that junction placement is superior to edge placement. Also, they presented a new integer-programming formulation for sensor placement in municipal water systems. In contrast to the Berry et al. [3] this model assumed that consumers are protected only if a sensor guards every path from a contaminant introduction site in order to avoid explicit references to time. The proposed model also assumed that flow within the pipes was reasonably swift, if not quantified, so that flow patterns will not shift before a contaminant is detected. Contaminant transportation is decoupled from sensor placement decisions and explicitly considers time. Because of the fact that EPANET is expensive in terms of computation, EPANET's hydraulic equations are simplified. Discrete-event simulation is used, via velocity information derived from EPANET, to calculate contamination timing in the network for each of possible number of attack scenarios. An integer program is used to select a set of sensor locations that minimizes the expected amount of contaminant delivered to consumers across all these attack scenarios.

Similar to Uber et al., we assumed that sensors are placed at only junctions in our methodology. Also we have considered the EPANET hydraulic simulation.

Watson et al. [33] differ from the literature in a way that they compared several different objectives: population exposed, time to detection, volume consumed, number of failed detections, and extent of contamination. They investigated the relationship between optimal placements obtained under different design objectives. Also they compared the sensor located edges and nodes, and they developed the mixed-integer linear programming models for the sensor placement problem over a range of design objectives.

However the related works include set-covering models that are solved by utilizing genetic algorithm or integer linear programming, Uber et al. [32] described a general formulation and a greedy heuristic solution methodology for the sensor location problem.

The aim of this work is to illustrate heuristic (non-optimal) solutions for large network, under uncertainty in threat scenario characteristics, and transport within the network.

Another associated paper, Berry et al. [5] categorized all of the mentioned sensor placement problems as static and dynamic. In static formulation, there are key assumptions such as contaminant dilution, concentration level and mode of attack is not modeled, and assumes identical contaminant and water flow dynamics. They claimed that these assumptions are not realistic, so they proposed a dynamic formulation that precisely characterizes the impact of an attack at a given network junction on the rest of the network. Because of the dynamic formulations require extensive computations, they proposed a heuristic solution, namely, GRASP (Greedy Randomized Adaptive Search Procedure) in order to generate quick solutions for larger networks.

There are similarities between our solution methodology and Berry et al. [5]. We use dynamic formulation, and in order to solve sensor location problems, we have proposed meta-heuristics instead of GRASP. However they solved the problem under minimum population exposed condition, we have tried to find the solution near the optimal according to both population exposure and time to detection objectives.

An additional paper, which are used the unsteady hydraulic conditions, is given by Propato et al. [26]. They formed nonlinear objectives with respect to population exposed, contaminant volume consumed, and time to detection. They proposed a polynomial programming formulation to detect random intrusions occurring at network nodes under the unsteady flow conditions in order to reduce nonlinear formulation to a mixed integer linear program. Compared to previous published works, this methodology aims to merge the simplicity of the problem formulations of two earlier studies, Berry et al. [3] and Watson et al. [33], both of which investigated the difficulty of solving some sensor-placement integer program on various computing stages.

Another issue that should be taken into consideration involves the uncertainties of the problem. While dealing with uncertainties in the sensor placement problem, Carr et al. [7] study's is a good starting point. They introduced two sensor placement objectives; one of them is minimize the expected population exposed and the other is to minimize the expected portion of the network that becomes contaminated as exactly we have. The paper considers the robust formulations for sensor placement that directly account for some uncertainties such as attack scenarios, water demands and the variability in the population density. In this paper, uncertainties are expressed as a random variable and deterministic MILP is used to minimize expected performance measure.

From the literature, obviously, in the case of external pollution, the time-to detection is the most important and should be considered in the design process. The second important issue is the minimization of the population exposure. Thereby, in this study the expected time to detection and expected population exposure are considered when presenting the optimal sensor location. The proposed methodology for finding the optimal solutions under various objectives shows important differences. The most crucial difference is the EPANET flow pattern. We illustrate a more realistic dynamic flow pattern than the already existing one in the literature. Another contribution is that literature does not consider the isolated nodes in the network. Isolated nodes are significant junctions, since if no sensors are located at those points, there is no chance to understand whether isolated nodes are contaminated or not. Due to this, we have located sensors at those kinds of nodes automatically, and we call those sensors as restricted sensors. Next, we have implemented a new meta-heuristics, which minimizes the expected detection time, and population exposure. Lastly, we have tried to show how the clustering affects the sensor location problem with proposed meta-heuristics.

3 SENSOR PLACEMENT IN WATER DISTRIBUTION SYSTEMS

In the literature, generally, the objective in a sensor placement optimization problem is the same: to place a limited number of sensors in water distributions network so that the impact of public health due to the accidental or targeted injection of contaminant is minimized. Commonly most important difference among the literature is how to design this impact parameter. Designing an impact parameter differs the objective functions according to the following five objectives proposed by Watson [33]:

- ✓ Population Exposed: The number of people that consumed contaminated water before detection.
- ✓ Time to Detection: The time between the attack and the detection.
- ✓ Volume Consumed: The amount of contaminated water that is consumed.
- ✓ Number of Failed Detection: The proportion of attacks that is undetected.
- ✓ Extent of Contamination: The length of pipe contaminated.

Apparently, sensor placement problem can be formulated as a location problem. Ostfeld et al. [21] solved detecting accidental contamination in municipal water networks problem by formulating it as a set-covering problem, which is closely related to location problems.

Moreover, we are inspired by the formulation of the problem presented by Berry et al. [4]. Note that Berry et al. [4] allow placement of at most p sensors and the objective function of the Berry et al. formulation is the minimization of the total consumption over all attacks with an optimal sensor configuration. A water network is modeled as a graph G=(V, E); E is the set of edge representing pipes; V is the set of vertices (or nodes) where pipe meets. Mathematical programming formulation of the problem is as follows:

$$\min \sum_{a \in A} \sum_{i \in La} \alpha_a w_{ai} b_{ai}$$

$$\text{s.t.} \sum_{i \in La} b_{ai} = 1, \quad \forall \text{a} \in A \ , \qquad (1)$$

$$\text{b}_{\text{ai}} \le s_i \ , \quad \forall \ \text{a} \in A, i \in L_a \ , \qquad (2)$$

$$\sum_{i \in L} s_i \le S_{\max} \ , \qquad (3)$$

$$b_{ai} \in \{0,1\}, s_i \in \{0,1\} \ \ \forall a \in A, \forall i \in L \ , \qquad (4)$$

where $L \subseteq V \times E$ is the set of possible sensor locations and $L_a \in V \times E$ is the set of network locations contaminated by attack a. A is the set of attacks. Parameters in the objective functions; α_a is the probability of an attack and w_{ai} (weight) the amount of contaminated water consumed at all vertices from an attack a to detection location i. Decision variable is s_i for each potential sensor location $\text{i} \in L$. This variable is 1 if we place a sensor at location i ; 0 otherwise. The set of constraints (1) enforces that there is exactly one best sensor for each attack scenario. The set of constraints (2) indicates the condition that detection can only occur at v_i if a sensor is placed there. The set of constraint (3) limits the total number of sensors to be placed ($S_{\max}$) [4].

3.1 Problem Definition

In this thesis, we decide to minimize expected time to detect and expected population exposed, since they are considered to be the most relevant and widely accepted in the literature. We also treated our problem as a location problem. Different from some of the literature, we assume that the probability of not detecting an attack equals to zero. For this purpose, we define *restricted sensor locations*. If the node is isolated, i.e., only inflow occurs to the node, we put sensors called restricted sensors to such special nodes. On the other hand, we also assume like most of the literature that the nodes are the possible sources of terrorist attacks. The mathematical formulations of the studied problems can be given as follows:

1) The first objective is the minimization of the expected time to detect;

$$\text{Minimize } 1/n \left(\sum_{i=1}^{n} C_i\right)$$

$$\text{s.t. } C_i = \min_j \{d_{ij}; X_j = 1\}, \quad (1)$$

$$\sum_{j=1}^{n} X_j = p, \quad (2)$$

$$X_j \in \{0,1\}, \quad (3)$$

where, the decision variable, X_j denotes whether a sensor is located at node j or not. The parameter d_{ij} is the shortest path in time units between the attack node i and node j, n is the total number of nodes in the network, p is the total number of sensors to be located, and i is the attack node. In this formulation, the objective function corresponds to minimizing the expected time to detect. Constraint (1) gives the minimum distance if the sensor is located at node j. Constraint (2) shows that exactly p sensors are available or there should be located. Constraint (3) ensures that whether sensor is located or not at node j.

2) The second objective is the minimization of the expected population exposed;

$$\text{Minimize } 1/n \left(\sum_{i}^{n} C_i\right)$$

$$\text{s.t } C_i = w_i \min_j \{d_{ij}; X_j = 1\}, \quad (1)$$

$$\sum_{j=1}^{n} X_j = p, \quad (2)$$

$$X_j \in \{0,1\}, \quad (3)$$

where parameter w_i is the demand at node i. In this formulation, the difference from the first objective is to multiply the objective function with the demand at node i. The aim is to minimize the population exposure with the same constraints of the first problem. Since, both of the models likely to be NP-hard, we limit our attention to heuristic solution methodologies.

3.2 Design Assumptions

For the sake of argument, the development of the methodology for an optimal detection system is based on the following concepts and assumptions. These assumptions do not change the nature of the problem, however they simplify the problem domain and ease of handling.

a. All quantities affecting the network model water quality predictions are assumed to be known and deterministic. The contaminant is stable after injection in finished drinking water. Each attack scenario involves a single injection location, which may occur at any network node and begin at any time with equal probability.
b. The systems are subjected to a 24-hour representative demand cycle. Pipe length, pipe diameter, tank elevation, tank diameter, initial, maximum and minimum water levels in the tanks are known.
c. For the purpose of calculating the expected population affected prior to detection, demand for each node and average travel time between nodes are known.
d. Sensors instantly detect any non-zero contaminant concentration and an action is taken to eliminate further exposure without delay.
e. EPANET inputs and networks to be studied are given.
f. EPANET version 2.00 is the standard model used to simulate the contaminant transport.
g. Regardless of pollutant concentrations, all the water that passes through contaminated water is assumed polluted.
h. Pollutant velocity is assumed to be the same velocity as the water velocity.
i. Contamination is made in a continuous mode.
j. Only one node becomes as source of pollution at a time.
k. Contaminant dilution within the water is neglected.
l. Sensors are only placed on nodes.
m. Without sensors in the network, a population at a node is exposed if contaminant can reach that point in a given flow period.

n. All the sensors in the network are identical.

All these stated assumptions on optimal sensor location problem are similar with the literature. Behind the problem definition and assumptions, now we provide the details of our methodology.

3.3 Design Methodology

Before presenting the details of the methodology, we have to discuss certain issues of the problem that require further attention and explanation.

1) Demand patterns usually change during 24-hours depending on the time of the day. During the daytime, business districts of the cities usually consume more water than the evening and midnight periods. On the other hand the residential neighborhoods usually demand more water during the evenings when the residents return to their homes after work. In the literature, this phenomenon is handled by dividing a 24-hour daily period into smaller time periods with different demand patterns. For example Ostfeld et al. proposed to work with three periods of 8-hours and different demand patterns associated with each time period at each node [21]. This assumption is reasonable and reflects the reality. However, it comes with a price in terms of computations. In order to minimize the computational cost, approximations can be done at different levels. In such an approximation effort, Ostfeld et al [21] proposed an auxiliary flow, termed *representative flow pattern* and explained it as follows:

Contamination due to an external intrusion is propagated by the flow pattern. Since an intrusion can occur at any given time and place, its propagation is possible by an infinite number of flow combinations. In order to handle this uncertainty, a single hypothetical flow pattern is created. This pattern, called a representative flow pattern is described by the time-averaged flow in each pipe and flow direction. The difference between flow directions is made on a network, where each pipe of alternated flow direction is substituted by two parallel pipes. For instance, suppose a 24-hour flow cycle of three different situations, 8 hour each, with corresponding flows at a specific pipe with the velocity of +1000, -700, and +1500 units. The pipe is represented on the network by two unidirectional and parallel pipes. The first pipe, appointed with a positive flow velocity of (1000*8/24)+

(1500*8/24)=833.3 units. The second pipe, assigned with a negative flow direction velocity of (700*8/24)=233.3 units. The contaminant propagation is so made possible in both flow directions, each with its corresponding time-averaged flow value [21]. If pipe length is assumed 1000 units, average time to travel is calculated as (1000/833.3)=1.20 units in positive direction and (1000/233.3)=4.30 units in negative direction. The resulting representative flow pattern is depicted in Figure 3.1

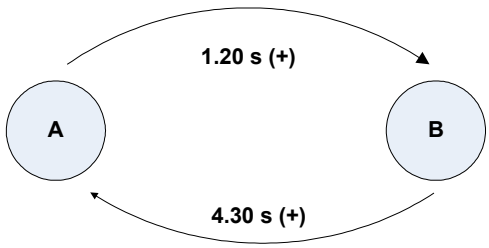


Figure 3.1 Representative Flow Pattern [21]

However, this approach has certain flaws while assuming bidirectional flow in the pipe at the same time. If a snapshot of the network is taken, evidently water flows unidirectional through the pipe. The approach adopted by Ostfeld et al. [21] creates nonexistent flows. Since, the representative flow pattern is calculated so that each pipe of alternated flow direction is replaced by two parallel pipes, one in each direction. Then the pipes are represented on the network by two undirectional and parallel pipes. Moreover, Ostfeld et al. [21] approach averages the flows that occur in a pipe throughout 24-hours. However this approach yields misleading conclusions. For example, if a pipe from node A to node B has a flow direction from node A to node B, between 8 AM and 4 PM time period, but the direction of the flow changes from node B to node A after 4 PM. Ostfeld's et al. [21] approach assumes that there is a certain (average) flow at both directions throughout the day. Based on this assumption a contaminant introduced from node A can be determined at node B (given that a sensor is located at node B) within an acceptable time determined by the average flow. On the other hand, in reality if the flow direction changes

after 4 PM, an intrusion cannot be detected until the direction of the flow changes. Therefore, the calculations based on the averaging technique of Ostfeld et al. [21] are misleading and should be corrected. In this study, calculation of the flow patterns is different than the calculations in Ostfeld et al. [21]. Briefly speaking, this nonexistent flow misleads the calculations and claims detection of certain attacks much earlier than what it would take in reality.

To overcome this issue, we propose to handle the flow patterns separately. For the above problem provided by Ostfeld, the corresponding flows' velocity are +1000 at 0-8 hour period, -700 at 8-16 hour period, and 1500 units at 16-24 hour period. And we have divided the flows' velocity to the pipe length in order to get the average travel time. If the pipe length is a 1000 unit, the average travel times for this particular pipe are (1000/1000)=1.00 unit at 0-8 hour period, (1000/700)=1.43 unit (negative direction) at 8-16 hour period, and (1000/1500)=0.67 unit at 16-24 hour period. The resulting flows with this approach is illustrated below in Figure 3.2

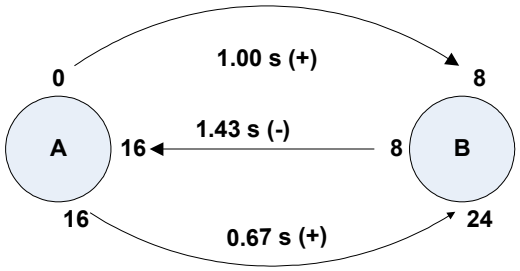


Figure 3.2 Proposed Flow Pattern

Handling the flows separately is a better approximation however requires further computation. For the above example, in order to estimate the travel time of the contaminant injected at node A, contaminant flows A through B in 1.00 hour between time period 0-8 hour. If the contaminant is injected in 8-16 time period, contaminant flows 4.67 hour. Since, injected contaminant at 8-16 time period has to wait the 16-24 time period which the flow turns out to be a positive direction. Assuming equal attack probability within the period, average weighting time of the contaminant in the node is half of the period, which is 4 hours. So, the time for the attack reaches to the node B is (4+0.67) =4.67 in the time period 8-16. Also, contaminant flows A through B in 0.67 hour between 16-24 time period. (Note that units are taken to be hours in order to represent the time)

2) While estimating the detection time of the contaminant, we have to first identify the nearest node with a sensor, which is in the *reachability set* of the attacked node. The reachability set of a particular node is the set of downstream nodes that the contaminant would flow through. The determination of the time to detect problem in that sense is associated with the shortest path problem. Note that in order to estimate the total consumption of the contaminated water before the contaminant is detected also requires the shortest path information among the nodes. The path with minimum length between two nodes on the auxiliary network is the shortest path. By realizing that each arc length equals to its averaged travel time (the pipe's physical length divided by its average velocity), the minimum propagation time from a source of pollution to any other node is equal to the shortest path between the source and the corresponding node.

3) Recall that we assumed that the probability of detection is assumed to be equal to 1. That is to say, every node in the network should be covered by at least one sensor. For this purpose we introduce a *restricted sensors placement* concept in our methodology. The restricted sensors placement is considered for inflow-only nodes, i.e., the nodes in which the flow occurs only into the node but not out (sink nodes). A contamination intrusion at such nodes can only be detected by a sensor that is placed at that node. We refer to these

sensors as *restricted sensors*.

4) A contaminant can be introduced to the network from any node. We assume that the source node for such an attack is distributed equally likely among the nodes of the network. Therefore, the estimation of the objective function values must consider all possible attack scenarios. Each node in the network is represented (covered) by its nearest sensor, which has the minimum average travel time to the nodes. For a given network, after determining possible contaminant intrusion scenarios, the calculation for the expected time to detection is done by summing the minimum average travel time to detect contamination for every possible node in the network to the nearest sensor and then dividing the total number of nodes in the network.

5) For the expected population exposure, the same procedure is followed. But in this case demand of a node is taken into consideration. In order to calculate how much contaminated water is consumed, demand of a node and minimum average travel time to pass detecting contamination by a sensor is multiplied. For the expected value of this, again this multiplication is divided into total number of nodes in the network.

Now we can provide the details of the proposed methodology.

3.4 Proposed Methodology

A methodology for finding the best set of sensor locations, subjected to a given number of sensors presented in this section. The flow of the methodology is depicted in Figure 3.3.

3.4.1 The Flow of the Methodology

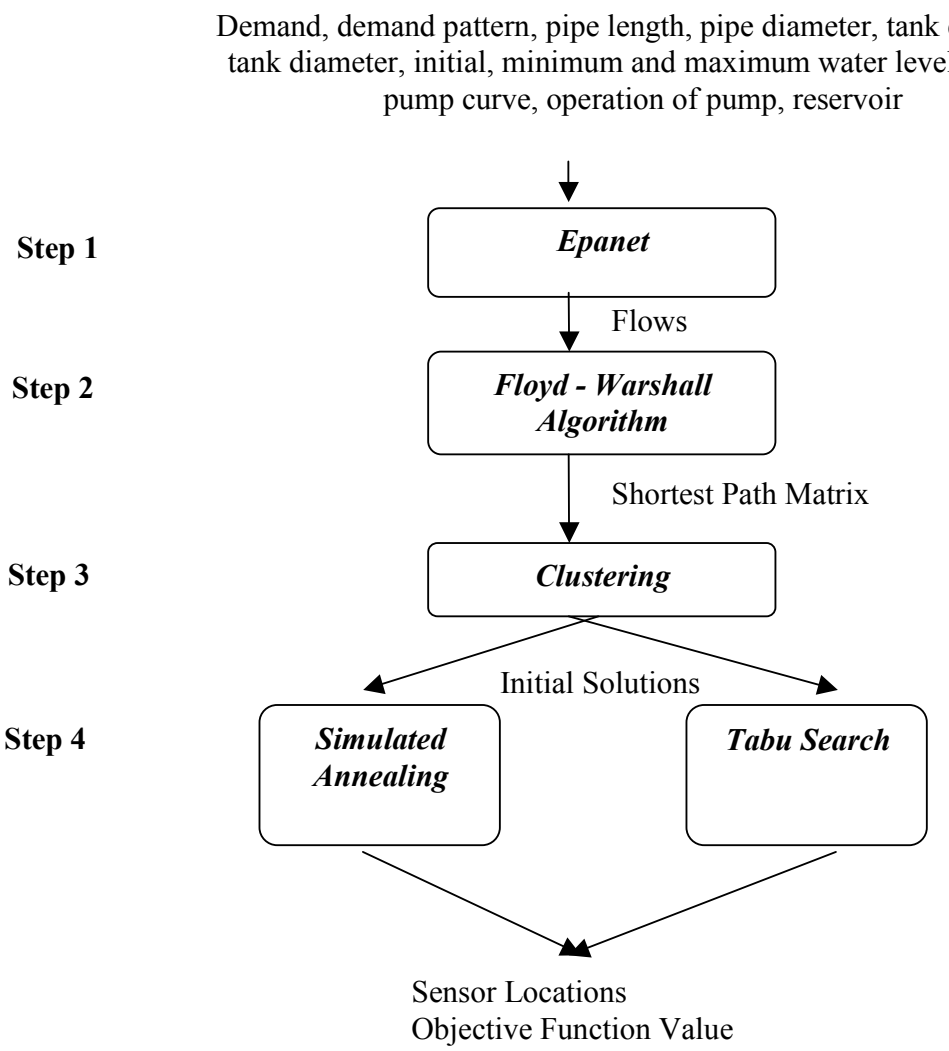


Figure 3.3 The Flow of the Methodology

3.4.2 Step 1

The proposed solution starts with a demand, pipe length, pipe diameter, tank elevations and levels, definition of pump curves and reservoirs input data which are water distribution systems elements data composed of sources, pipes and hydraulic control elements such as valves, tanks, and reservoirs. In order to identify the mechanic of the problem network that is mainly flows and velocities, we used EPANET introduced by Rosmann [28], which is designed to be a tool for improving our understanding of the movement and fate of drinking water constituents within distribution systems.

EPANET gives a solid background for the extended-period simulation of the water distribution network. A network consists of pipes, nodes (pipe junctions), pumps, valves and storage tanks or reservoirs. EPANET tracks the flow and velocity of water in each pipe, the pressure at each node, the height of water in each tank, and the concentration of a chemicals throughout the network during a simulation period included of multiple time steps. The simulation covers a typical demand cycle, usually one-day or one week. The output of the hydraulic simulation to be used comprises the time history of the flow velocities for each pipe in the network.

3.4.3 Step 2

After we identify the flows via EPANET for the given demand pattern and the network structure, next step is determining the shortest path distances between the nodes. For this purpose we utilized Floyd-Warshall [1] algorithm. Before presenting the details of the Floyd-Warshall algorithm we will briefly introduce the shortest path problem and the associated terminology.

Note that, the flows obtained from the EPANET, assists us to generate a directed graph representation of the water distribution network, which is composed of a set of nodes and a set of directed arc. The set of nodes equals to the network's original set of nodes. The

set of arc is allocated, based on the simulation results, as follows. A directed arc e (i, j) is assigned if there exists a pipe between node i and node j and if flow occurs only from node i to node j.

The length of an arc corresponds to the average travel time along the arc, which is equal to the physical length of the corresponding arc divided by its average flow velocity.

Assuming that the pollutant moves with a velocity equal to the flow velocity, the time of the pollutant's propagation from node i to node j is equal to the minimum travel time between these two nodes. Hence, the solution to the all-shortest paths problem identifies the propagation times from any given source of pollution to all other nodes. Based on the average travel time data of each arc obtained from EPANET, Floyd Warshall [1] algorithm was implemented in order to identify the all-shortest paths in the network. Therefore, distance matrix is formed for the next step.

3.4.4 Step 3

After we determine the shortest path, we apply a clustering algorithm in order to identify a good initial solution. Note that, a good starting solution usually leads to better solutions while working with local search heuristics such as Tabu Search and Simulated Annealing.

Clustering is grouping of data objects with high degree of similarity in the same cluster while placing those that are dissimilar to other clusters. The goal of clustering is to determine the intrinsic grouping in a set of unlabeled data.

In this study, k-medoids clustering algorithm is implemented in order to get initial solution for the sensor placement. The two dimensional distance matrix formed by Floyd Warshall algorithm [1] is utilized for this purpose.

The k-medoids algorithm is based on the search for k representative objects or medoids among the observations of the dataset. These observations should represent the structure of the data, that is to say, they should be the prototypes of the corresponding clusters. After finding a set of k medoids, k clusters are formed by assigning each observation to the nearest medoid.

The aim of the algorithm is to find k representative cluster center, which minimize the sum of the dissimilarities of the observations to their closest representative cluster center. By default, when medoids are not specified, the algorithm first looks for a good initial set of medoids that is cluster centers. Then it finds a local minimum for the objective function that will decrease the objective called the swapping [14].

In this study, we utilized R project tool [27] developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues for clustering in order to get medoids. Next, the medoids are accepted as the initial sensor locations for our problem.

In R project, the k-medoids routine performs k-medoids clustering on a given input, using the distance matrix and the number of clusters passed by the user. Multiple passes are being made in order to find the optimal clustering solution, each time starting from a different initial clustering. The following illustrative figure shows how the kmedoids technique works.

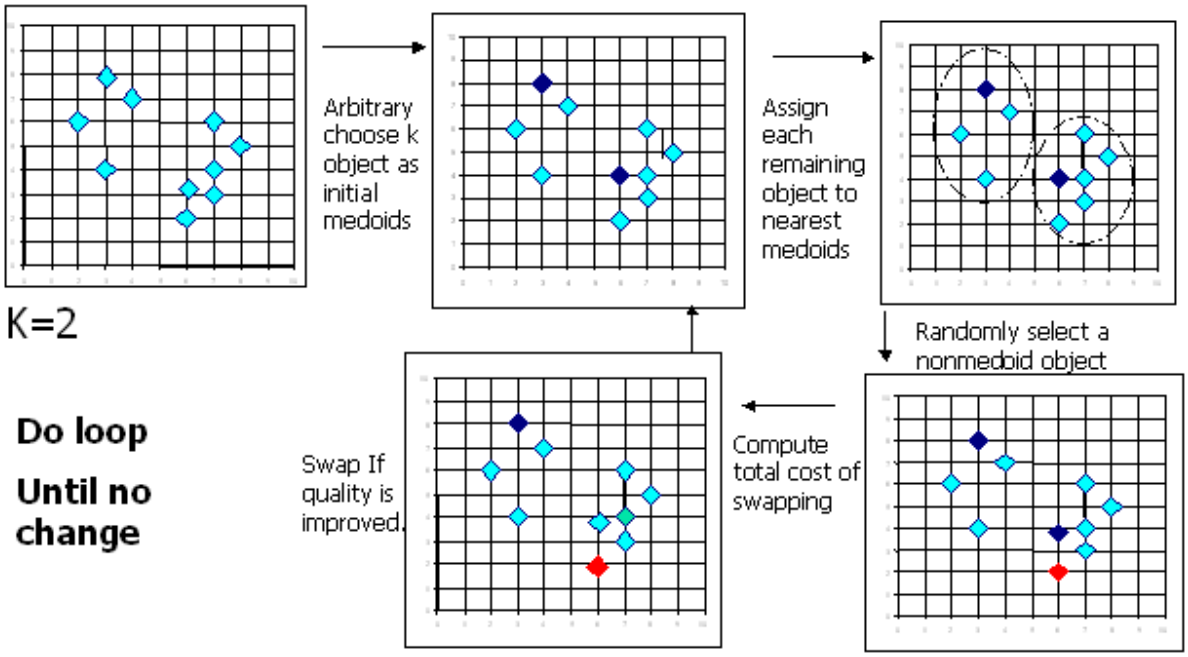


Figure 3.4 A Representative Example for the kmedoids Technique

3.4.5 Step 4

After obtaining the initial sensor locations, metaheuristics are utilized in order to solve the problem. We propose two metaheuristics, namely simulated annealing and tabu search at this step.

3.4.5.1 Simulated Annealing

Metropolis et al. [20] first introduced Simulated Annealing (SA) also known as Monte Carlo annealing in 1953. Simulated annealing method initiated from the Annealing process of thermodynamics, where a solid achieves thermal equilibrium after being frozen slowly under gradually decreasing temperatures. The crucial advantage of simulated annealing is its ability to escape local optimum.

The *Annealing*, is a process for obtaining low energy states of the substance. That means substance rose to higher energy levels by melting, then it slowly cooled until it becomes solid. The aim of this process is to get the minimal energy state of the substance [13].

Simulated annealing is a widely recognized optimization tool. According to Juraj Hromkovic [31] SA is successful because; 1) It is based on a simple idea and can be implemented easily 2) Very robust and can be applied to a wide range of optimization problems 3) The randomness included in the mechanism allows it to outperform local search.

Evidently, the choice of parameters of the simulated annealing design (the neighborhood and the cooling schedule) is among the key elements assisting the success of the application of the simulated annealing algorithm.

Simulated Annealing proceeds by perturbing a possible solution or state by moving to another state in its neighborhood. The objective function is calculated for the latest state

and compared to the objective function of the earlier state. In minimization, if the latest state has a lower objective function than its predecessor, the latest state is accepted replacing accordingly the older one, and mainly a step downwards is taken towards the global or local minimum. In contrast, if the latest state's performance is worse than its predecessor, simulated annealing may still accept the new worse state based on a probabilistic model, that is a step upwards may be permitted in a minimization problem.

The simulated annealing requires an initial temperature T, in order to begin with and a cooling schedule. The initial value T has to be huge enough to let moves to be accepted, because it should correspond to heating the solid until all particles are arbitrarily arranged in the liquid stage. While choosing the temperature reduction, the typical method for the temperature reduction is to multiply T by some constant r, $0.8 \le r \le 0.99$. Using this reduction one works a constant number k of iterative steps with a fixed T, and after k steps $T = r \times T$.

In this thesis, we propose two different neighborhood generation methods and compare their performance both in terms of objective function values and computational requirements. In the first method, the initial sensor configuration is selected among the network nodes, and at each subsequent step, a candidate solution is constructed randomly, that is to say the neighborhood generation is random. If certain conditions are met, the candidate solution becomes the new current solution and repeats the same procedure until a termination criterion is met. We refer to this scheme as Randomized Simulated Annealing. Note that this method is basically a random search algorithm with a Simulated Annealing flavor during the new current solution selection process.

The second neighborhood generation method is a more conventional approach, in which a set of candidate solutions is created with respect to a more structured neighborhood generation criterion. We refer to this neighborhood search mechanism as Structured Simulated Annealing. In this research we adopt a one-swap neighborhood generation approach. In this approach, we select a node from the set of nodes in which a sensor is placed at the current solution and swap it with a node, which is adjacent (has a direct arc in-

between) to it. All possible swaps lead to a set of candidate solutions, which are defined to be the neighbors of the current solution.

The objective function is utilized as the fitness function and selection criteria during evaluating the performance of the candidate solutions. If the fitness function value of the candidate state is better than the current solution, the candidate solution is accepted as the new current solution apart from the temperature value. On the other hand if the performance of the candidate solution is worse than its predecessor, first a random number p denoting a random probability is calculated from a uniform random distribution on the interval [0, 1]. Then the following Metropolis criterion is checked choosing whether to accept the new state or not;

$$p = e^{\frac{-\Delta E}{T}},$$

where T represents the actual temperature and ΔE represents the difference between the fitness value of the candidate solution and the current solution:

ΔE = Fitness (Candidate Solution) – Fitness (Current Solution)

It is easily seen that, in the case of a minimization problem, a state is better than another state if its fitness value is lower than the other state's fitness value. The algorithm illustrates a basic simulated annealing process of a minimization problem.

algorithm Simulated Annealing

Initialize Temperature T_0 where $T_0 > 0$

Select an Initial Solution I_0

Set $k = 1$ and $T_k = T_o$

Set Current Solution as an Initial Solution $CS_k = I_0$

While Stopping Criterion > T_k

Repeat Until the Termination Criteria is met (e.g., for a certain number of iterations N)

Select a new point in the neighborhood of the, CS_k namely NS_k

Evaluate $\Delta E = \Delta E_{CS_k} - \Delta E_{NS_k}$

If $\Delta E \leq 0$ then set $CS_k = NS_k$

Else set $CS_k = NS_k$ with a probability of $e^{\frac{-\Delta E}{T_k}}$

Set $k = k + 1$

Set $T_k = rT_{k=1}$

End

Figure 3.5 Algorithm for the Simulated Annealing Technique

In this thesis a second meta-heuristic, namely the Tabu Search Algorithm is also implemented for the sensor placement problem. Next we provide the details of the proposed Tabu Search Algorithm.

3.4.5.2 Tabu Search

Tabu search is a heuristic based on local search. It is a relatively new technique first proposed in 1977 by Glover [10]. The major advantage of Tabu Search is its ability to escape indefinite cycling conditions. The idea behind this ability is that, there is no possibility of accepting a bad solution or step except it is going to assist to escape the cycling situations by preventing a path already visited. Thus, tabu search techniques require an internal memory, which grasps the paths already visited, i.e., tabu list. Note that, simulated annealing and local search algorithms are memoryless, that is to say moving to a candidate solution depends solely on the current solution. However, tabu list allows utilizing certain number of previous solutions to be effective in the selection process [13].

The Tabu list, usually executed as an array, which forbids some paths, most likely the previously visited ones, which eventually gets rid of the risk of oscillation or cycling. And this is the idea of the tabu search: to store some information about a sequence of the last feasible solutions generated and to use this information when generating the next feasible solution.

Tabu search algorithm is also a neighborhood search algorithm, thus requires a neighborhood generation mechanism. In this study, the two neighborhood generations methods discussed above are utilized.

While designing a Tabu search algorithm the Tabu list, its organization, its length and implementation should have been taken into consideration. The Tabu list's length should be cautiously considered, as an incorrect choice of the list's length may yield an ineffective algorithm. It is discussed that the list length between 7 and 15 is generally appropriate for a wide range of applications [30].

```
algorithm Tabu Search
Identify initial Solution
Create empty TabuList
Set BestSolution =Solution
Define TerminationConditions
Done=false
Repeat
        If Value of Solution > Value of BestSolution Then
        BestSolution=Solution
                If no TerminationConditions have been met yet
                Then
                Generate all neighbors and select the best one
                Select best non Tabu neighbor as next solution
                Add Solution to TabuList
                        If TabuList is full then
                        Delete oldest entry from TabuList
        If no non tabu neighbor solution was found OR
        If no improved has been achieved on best objective value
        Then
        Done=True
        Else
                Assign NewSolution as a neighboring
                If New Solution not on Tabu List Then
                Solution = New Solution
End
```

Figure 3.6 Tabu Search Pseudocode

4 COMPUTATIONAL RESULTS and EXPERIMENTAL ANALYSIS

In this chapter, we provide the details of the experimental design and compare the performances of the proposed algorithms both in terms of the objective function values and computational requirements. The proposed algorithms are coded in C++ language using Microsoft Visual Studio 6.0.

We first provide the details of the experimental analysis. Later we provide the results of the analysis. In this research we compared the performances of the proposed algorithms both in terms of the objective function values and computational costs. We also analyzed the performance of the algorithms under various parameter settings and subroutines such as the two neighborhood generation mechanisms discussed in the previous chapter and utilizing or skipping the k-medoids clustering algorithm while generating the initial solution.

4.1 Experimental Conditions

There five experimental factors that can affect the efficiency of the proposed algorithms.

These factors are listed as follows:

1. Size of the water distribution networks
2. Number of Sensors located

In this thesis we utilized four water distribution networks that are available in the literature as benchmark problems. The first one is a small network with only 9 nodes. The second and third networks are medium sized with 33 and 92 nodes respectively. The fourth water distribution network is a representative of large networks with 1192 nodes. (Table 4.1)

We also considered three different density levels in terms of the number of sensors that would be located. These three levels will be referred to as low density, medium density and high density. Note that, the number of sensors to be located not only depends on how dense the network would be but also to the size of the network and number of restricted nodes in the network. Therefore we came up with the following table, which illustrates the number of sensors for each density category and network size.

Table 4.1 Size of the Network and Density Category of Sensors

	Number of Nodes in the Distribution Network (Size of the Network)			
DC # of Nodes	9	33	92	1129
LD # of sensors	2	7	19	334
MD # of sensors	3	10	27	422
HD # of sensors	4	13	35	510

* DC: Density Category

* LD: Low density, MD: Medium Density, HD: High Density

4.2 Variations in the Algorithm

We also investigated the performance of the algorithms with respect to some variations in the methodology. These variations are listed as follows;

1. Neighborhood Generation Mechanism: Structured or Randomized
2. Selecting the initial solution based on clustering or not

Recall that the Structured Neighborhood generation corresponds to selecting a node from the set of nodes in which a sensor is placed at the current solution and swap it with a node, which is adjacent (has a direct arc in-between) to it. On the other hand, in Randomized Neighborhood generation, at each subsequent step, a candidate solution is constructed randomly.

The performance of the local search algorithms usually improves if the search starts from a good initial solution. In order to determine the effects of such a strategy, we decided to include a clustering stage, which can provide a good initial solution to the meta-heuristics.

4.3 Parameters of the Algorithm

One of the major disadvantages of meta-heuristics is the number of parameters that should be tuned for a good solution. Both simulated annealing and tabu search algorithms have relatively low number of parameters compared to other meta-heuristics such as genetic algorithms. Now we discuss the parameter selection process implemented in this research.

4.3.1 For the Simulated Annealing (SA)

In this thesis, we set the initial temperature of the SA as the initial objective function value, which is large enough for initial temperature. The cooling parameter r is chosen as 0.99, because of the nature of the problem [29]. Third parameter that should be decided is the termination criterion. We adapted an approach, which checks two conditions. First condition implies to terminate the algorithm as soon as the temperature drops below a certain level. We compare 5 different levels according to their performances. Note that as the level decreases, the computational time increases. The terminating temperature levels for experimental analysis are chosen as 0.1, 0.01, 0.001, 0.0001, and 0.00001. The second condition is a conventional no changes for some time then stop criterion. That is to say, the

algorithm terminates when the objective function value does not improve for the last 50 iterations.

4.3.2 For the Tabu Search

Among the parameters for the Tabu Search, *tabu list* is the most significant one. Therefore, we opt to analyze the performance of the Tabu search algorithm with various *tabu list* sizes. We set of the size of the tabu list to 5, 7, 10 and 15, and compared the performance of the Tabu Search algorithm [30]. Note that, if the size of the tabu list is too large, one can not generate neighbors, that is to say the *intensification* objective of a local search algorithm suffers. On the other hand, a short tabu list may prohibit a *diverse* search. For the TS algorithm, we utilize a termination criterion similar to the latter condition adopted in simulated annealing. According to this termination criterion, the algorithm stops if the objective function value does not change for the last 10 iterations.

After introducing the experimental conditions and parameters of the problem, next we present and discuss the results.

4.4 Results

In Table 4.2, the expected times to detect values are presented for the SA algorithm with clustering and SA algorithm without clustering with respect to the structured neighborhood search. In the same table, the objective function values corresponding to different temperature level parameters also presented. It seems that as the size of the network increases both the effect of clustering and the temperature level parameter becomes more significant.

Table 4.2 Expected Time to Detect with Structured Neighborhood Search for SA

		SASC without Clustering					SASC with Clustering				
A	**B**	0.1	0.01	0.001	0.0001	0.00001	0.1	0.01	0.0001	0.00001	0.000001
	2	3.94	3.94	3.94	3.94	3.94	3.94	3.94	3.94	3.94	3.94
	3	2.51	2.51	2.51	2.51	2.51	2.51	2.51	2.51	2.51	2.51
9	4	1.62	1.62	1.62	1.62	1.62	1.62	1.62	1.62	1.62	1.62
	7	7.64	7.64	7.64	7.64	7.64	8.23	7.64	7.64	7.64	7.64
	10	5.44	4.47	4.47	4.47	4.47	5.44	4.31	4.31	4.31	4.31
33	13	3.47	2.32	2.32	2.32	2.32	4.51	3.92	3.53	3.53	3.53
	19	2.88	2.17	1.86	1.86	1.86	2.67	2.14	1.88	1.88	1.88
	27	1.64	1.26	0.96	0.96	0.96	1.61	1.34	1.00	1.00	1.00
92	35	1.06	0.65	0.52	0.52	0.52	1.10	0.83	0.61	0.61	0.61
	334	2269.30	2217.83	2073.47	1914.06	1890.02	1854.54	1854.54	1836.15	1616.90	1616.90
	422	1792.28	1792.28	1496.19	1278.26	1122.37	1522.70	1522.70	1504.09	1302.09	1273.29
1129	510	1325.07	1296.71	1166.63	891.68	761.97	1108.08	1108.08	1070.42	871.10	786.89

* SASC: Simulated Annealing Stopping Criteria

From the tabulated results, as easily can be seen that if the density of the sensors increase the stopping criteria play an important role for the results. Moreover, in small sized networks, we could not observe significant change under the different stopping criteria both for the clustering and without clustering cases.

On the other hand, in the large network, all the parameters of stopping criteria alter the results. And we may conclude that clustering is a good alternative for large networks in simulated annealing with structured neighborhood search in order to minimize the expected time to detect.

In Table 4.3, the expected population exposure values are presented for the SA algorithm with clustering and SA algorithm without clustering with respect to the structured neighborhood search. In the same table, the objective function values corresponding to different temperature level parameters also presented.

Table 4.3 Expected Population Exposed with Structured Neighborhood Search for SA

		SASC without Clustering					SASC with Clustering				
A	**B**	0.1	0.01	0.001	0.0001	0.00001	0.1	0.01	0.0001	0.00001	0.000001
	2	958.15	958.15	958.15	958.15	958.15	958.15	958.15	958.15	958.15	958.15
	3	400.25	400.25	400.25	400.25	400.25	400.25	400.25	400.25	400.25	400.25
9	4	192.96	192.96	192.96	192.96	192.96	192.96	192.96	192.96	192.96	192.96
	7	174.66	174.66	174.66	174.66	174.66	164.42	164.42	164.42	164.42	164.42
	10	102.76	69.76	69.76	69.76	69.76	97.90	78.62	78.62	78.62	78.62
33	13	59.14	34.58	34.58	34.58	34.58	61.39	38.04	33.58	33.58	33.58
	19	244.22	137.30	123.02	123.02	123.02	479.76	350.32	118.73	118.73	118.73
	27	154.56	53.55	29.83	29.83	29.83	163.81	102.73	42.29	37.90	37.90
92	35	52.27	21.88	12.35	12.35	12.35	79.06	59.95	31.86	28.47	28.47
	334	45536.90	42179.50	37345.90	28776.10	24295.60	30556.30	30556.30	30556.30	27614.00	26430.10
	422	28184.40	26128.10	20474.10	15676.00	11718.10	27393.10	27393.10	24881.30	21852.60	20522.60
1129	510	16810.50	15472.70	12549.30	10136.20	6352.22	18299.90	18299.90	16689.50	14062.70	12765.90

Here, one can easily realize that without clustering superior to clustering. While implementing clustering, in the minimization of the expected population exposed, we have only considered the distance matrix. On the other hand, both distance matrix and demand patterns should have been taken into consideration.

In Table 4.4, the expected times to detect values are presented for the Tabu Search algorithm with clustering and Tabu Search algorithm without clustering with respect to the structured neighborhood search. In the same table, the objective function values corresponding to different tabu size parameters also presented. We have realized that as the size of the tabu list increases, objective values does not change. So we have concluded that size of the tabu list does not have significant affect on the solution.

Table 4.4 Expected Time to Detect with Structured Neighborhood Search for Tabu

		TSP without Clustering				**TSP with Clustering**			
A	**B**	5	7	10	15	5	7	10	15
	2	3.94	3.94	3.94	3.94	3.94	3.94	3.94	3.94
	3	2.51	2.51	2.51	2.51	2.51	2.51	2.51	2.51
9	4	1.62	1.62	1.62	1.62	1.62	1.62	1.62	1.62
	7	7.64	7.64	7.64	7.64	7.64	7.64	7.64	7.64
	10	4.47	4.47	4.47	4.47	5.04	5.04	3.89	5.10
33	13	2.50	2.37	2.50	2.50	3.90	3.90	3.95	2.56
	19	1.86	1.86	1.86	1.86	2.02	1.84	1.89	1.78
	27	0.94	0.93	0.88	0.88	1.07	1.14	0.92	0.94
92	35	0.47	0.48	0.44	0.44	0.47	0.47	0.47	0.47
	334	1770.60	1770.60	1770.17	1766.74	1567.17	1565.00	1551.04	1548.81
	422	1175.63	1175.63	1165.68	1128.16	1186.53	1185.88	1183.23	1182.87
1129	510	699.24	699.24	698.97	694.20	789.90	789.61	787.44	787.44

*TSP: Tabu Search Parameter

Tables are tabulated in order to investigate the relationship between tabu size and our objectives. In the small networks, we have shown that length of the tabu size does not affect the results so much. In moderate sized network, the tabu size length could be 7 to 10. In the large network, it is proper to take the tabu size length as 15.

In Table 4.5, the expected population exposed values are presented for the Tabu Search algorithm with clustering and Tabu Search algorithm without clustering with respect to the structured neighborhood search. In the same table, the objective function values corresponding to different tabu size parameters also presented. In fact as the size of the network increases, there is not much considerable change both the effect of clustering and the tabu size length.

Table 4.5 Expected Population Exposed with Structured Neighborhood Search for Tabu

		TSP without Clustering				TSP with Clustering			
A	**B**	5	7	10	15	5	7	10	15
	2	958.15	958.15	958.15	958.15	958.15	958.15	958.15	958.15
	3	567.99	567.99	567.99	567.99	646.52	646.52	646.52	646.52
9	4	213.05	213.05	213.05	213.05	213.05	213.05	213.05	213.05
	7	383.52	383.52	383.52	383.52	368.70	368.70	376.01	383.74
	10	108.32	108.32	108.32	108.32	305.75	305.75	305.75	305.75
33	13	63.79	63.79	63.79	63.79	251.88	251.88	257.05	258.15
	19	660.20	660.20	660.20	660.20	528.07	528.07	546.92	769.02
	27	48.38	48.38	48.38	48.38	113.27	110.20	110.20	108.95
92	35	21.54	21.70	21.95	22.61	44.75	45.44	44.75	43.13
	334	35039.00	35039.00	35039.00	35039.70	27072.60	27072.60	27071.70	27056.80
	422	21442.00	21442.00	21442.00	21442.00	22061.10	22061.10	22116.90	22117.20
1129	510	9958.68	9958.68	9958.68	9909.59	15258.70	15258.70	15219.20	15204.20

In Table 4.6, the expected time to detect values is tabulated for the SA algorithm with clustering and SA algorithm without clustering with respect to the randomized neighborhood search. In the same table, the objective function values corresponding to different temperature level parameters also presented.

Table 4.6 Expected Time to Detect with Randomized Neighborhood Search for SA

		SASC without Clustering					SASC with Clustering				
A	**B**	0.1	0.01	0.001	0.0001	0.00001	0.1	0.01	0.001	0.0001	0.00001
	2	3.94	3.94	3.94	3.94	3.94	3.94	3.94	3.94	3.94	3.94
	3	2.51	2.51	2.51	2.51	2.51	2.51	2.51	2.51	2.51	2.51
9	4	1.62	1.62	1.62	1.62	1.62	1.62	1.62	1.62	1.62	1.62
	7	7.73	7.64	7.64	7.64	7.64	7.73	7.64	7.64	7.64	7.64
	10	4.96	3.91	3.91	3.91	3.91	5.75	3.91	3.91	3.91	3.91
33	13	2.76	2.37	2.37	2.37	2.37	4.08	2.56	2.37	2.37	2.37
	19	2.33	1.77	1.72	1.72	1.72	2.45	1.99	1.77	1.77	1.77
	27	1.54	1.08	0.99	0.99	0.99	1.60	1.22	1.05	1.05	1.05
92	35	1.05	0.67	0.53	0.53	0.53	1.06	0.53	0.51	0.51	0.51
	334	2049.67	2049.67	1539.36	1300.00	1300.00	1855.12	1855.12	1518.86	1352.61	1352.61
	422	1611.39	1565.41	1247.31	844.35	824.49	1518.32	1475.91	1265.36	888.48	888.48
1129	510	1120.70	1120.70	1009.41	643.67	542.65	1110.27	1110.27	838.90	548.34	548.34

As in the case of SA with structured neighboring, in small sized networks, we could not observe significant change under the different stopping criteria both for the clustering and without clustering cases. However, stopping criteria is a significant parameter in the large network. We have observed that for large networks if we allow more iteration, clustering is better than without clustering.

In Table 4.7 the expected population exposed values are depicted for the SA algorithm with clustering and SA algorithm without clustering with respect to the randomized neighborhood search. In the same table, the objective function values corresponding to different temperature level parameters also presented.

Table 4.7 Expected Population Exposed with Randomized Neighborhood Search for SA

		SASC without Clustering					SASC with Clustering				
A	**B**	0.1	0.01	0.001	0.0001	0.00001	0.1	0.01	0.001	0.0001	0.00001
	2	958.15	958.15	958.15	958.15	958.15	958.15	958.15	958.15	958.15	958.15
	3	400.25	400.25	400.25	400.25	400.25	400.25	400.25	400.25	400.25	400.25
9	4	192.96	192.96	192.96	192.96	192.96	192.96	192.96	192.96	192.96	192.96
	7	174.68	164.59	164.59	164.59	164.59	164.42	164.42	164.42	164.42	164.42
	10	83.50	72.35	69.76	69.76	69.76	97.35	71.31	69.76	69.76	69.76
33	13	63.04	37.84	37.84	37.84	37.84	59.17	48.62	33.29	33.29	33.29
	19	236.49	122.74	122.74	122.74	122.74	287.40	197.67	147.84	147.84	147.84
	27	95.37	59.02	31.60	31.60	31.60	108.89	85.91	32.69	32.69	32.69
92	35	42.87	17.54	11.91	11.91	11.91	56.00	21.11	13.52	13.52	13.52
	334	34951.00	28327.40	20074.50	15384.40	15384.40	30556.30	30556.30	18654.40	15196.40	15196.40
	422	22900.40	18423.10	11275.10	7555.76	7441.05	20543.90	17531.30	13073.10	7626.86	7369.08
1129	510	13518.50	11717.40	8202.62	5020.96	3987.20	13051.70	11124.90	7946.28	5078.10	4176.89

For all of the discussions about simulated annealing with or without clustering, randomized or structured neighboring search, we might generalize that results between the stopping conditions for the small, moderate as almost the same as different parameters. But in the large network, we can realize the effect of parameters.

In tabulated results, the expected time to detect and expected population exposed values are illustrated for the Tabu Search algorithm with clustering and Tabu Search algorithm without clustering with respect to the randomized neighborhood search.

Table 4.8 Expected Time to Detect with Randomized Neighborhood Search for Tabu

		TSP without Clustering				TSP with Clustering			
A	**B**	5	7	10	15	5	7	10	15
	2	3.94	3.94	3.94	3.94	3.94	3.94	3.94	3.94
	3	2.51	2.51	2.51	2.51	2.51	2.51	2.51	2.51
9	4	1.62	1.62	1.62	1.62	1.62	1.62	1.62	1.62
	7	7.64	7.64	7.64	7.64	7.64	7.64	7.64	7.64
	10	3.89	3.89	3.89	3.89	3.89	3.89	3.89	3.89
33	13	2.32	2.32	2.32	2.32	2.32	2.32	2.32	2.32
	19	1.72	1.72	1.72	1.72	1.72	1.72	1.72	1.72
	27	0.86	0.86	0.86	0.86	0.86	0.86	0.86	0.86
92	35	0.44	0.44	0.44	0.44	0.44	0.44	0.44	0.44

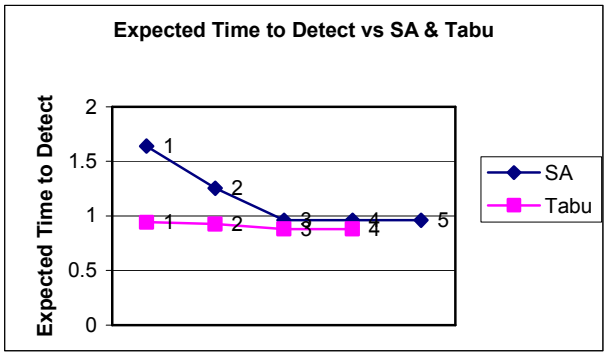
Table 4.9 Expected Population Exposed with Randomized Neighborhood Search for Tabu

		TSP without Clustering				TSP with Clustering			
A	**B**	5	7	10	15	5	7	10	15
	2	958.15	958.15	958.15	958.15	958.15	958.15	958.15	958.15
	3	400.25	400.25	400.25	400.25	400.25	400.25	400.25	400.25
9	4	192.96	192.96	192.96	192.96	192.96	192.96	192.96	192.96
	7	164.42	164.42	164.42	164.42	164.42	164.42	164.42	164.42
	10	69.76	69.76	69.76	69.76	69.76	69.76	69.76	69.76
33	13	32.88	32.88	32.88	32.88	32.88	32.88	32.88	32.88
	19	106.48	106.48	106.48	106.48	106.48	106.48	106.48	106.48
	27	28.63	28.63	28.63	28.63	28.87	28.63	28.63	28.63
92	35	10.61	10.61	10.61	10.61	10.61	10.61	10.61	10.61

The results show that randomized search in different Tabu size length does not affect the solution. Also, the large network solutions could not get, because of the high computational time of randomized neighborhood search. Since, in the randomized search case, the algorithm search for the all neighbors apart from current node. Because of this reason, tabu search with randomized neighborhood search is not efficient for large network.

In order to realize which techniques, stopping criteria (1,2,3,4,5) that is (0.1, 0.01, 0.001, 0.0001, 0.00001) or length of tabu size (1,2,3,4) that is (5, 7, 10, 15) are superior, we have analyzed the results. The following figures illustrate the best-expected time to detect and population exposed values for the different parameters in design case 3 with 27 sensors located through the network.

1. Experiment for the Structured Neighborhood Search for the SA vs Tabu



*ETD: Expected Time to Detect Values

Figure 4.1 ETD for Structured Neighborhood without Clustering

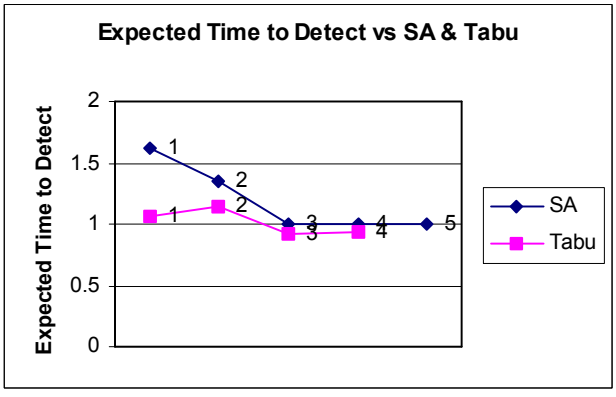
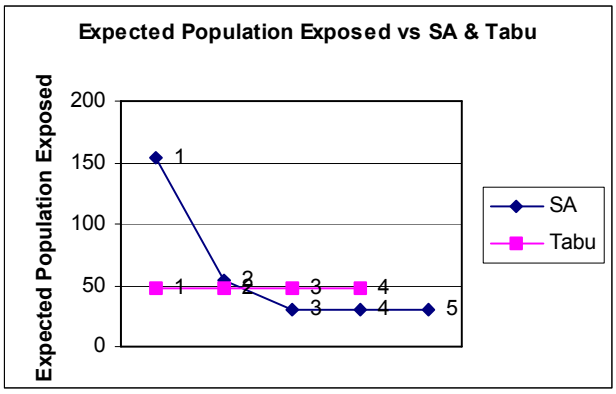


Figure 4.2 ETD for Structured Neighborhood with Clustering

Tabu noticed the solution that is nearer to best solution at the first iteration, however the simulated annealing should iterate at least 2 times in order to reach the closer result. Also, Length of tabu size does not have significant affect on the best solution for the minimum expected time to detect. In the clustered case, again the conclusion is same as the other parameters.



*EPE: Expected Population Exposed

Figure 4.2 EPE for Structured Neighborhood without Clustering

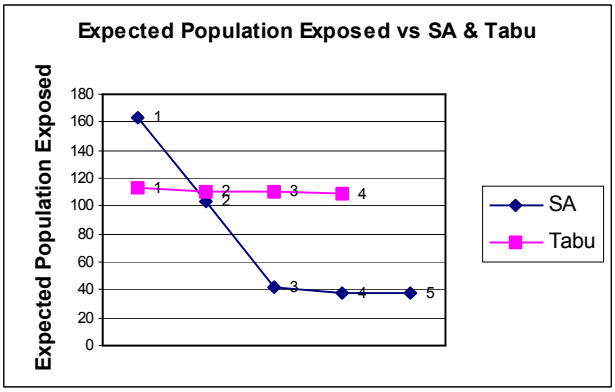


Figure 4.4 EPE for Structured Neighborhood with Clustering

For the expected population exposed, simulated annealing gives better solution than tabu search and SA reaches the best solution at the third stopping criterion. Also, it is easily say that the results for the different tabu size length are the same.

2. Experiment for the Randomized Neighborhood Search SA versus Tabu

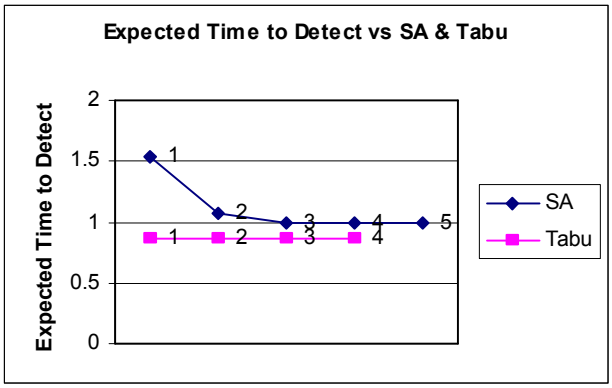


Figure 4.5 ETD for Randomized Neighborhood without Clustering

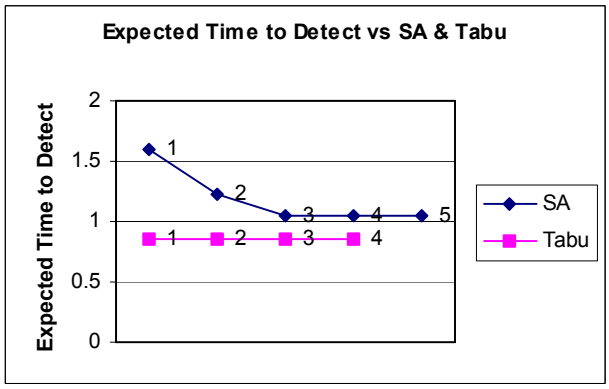


Figure 4.6 ETD for Randomized Neighborhood with Clustering

For the Randomized neighborhood search tabu search presents a better solution than simulated annealing. Also, tabu size doesn't influence the solution for the randomized case.

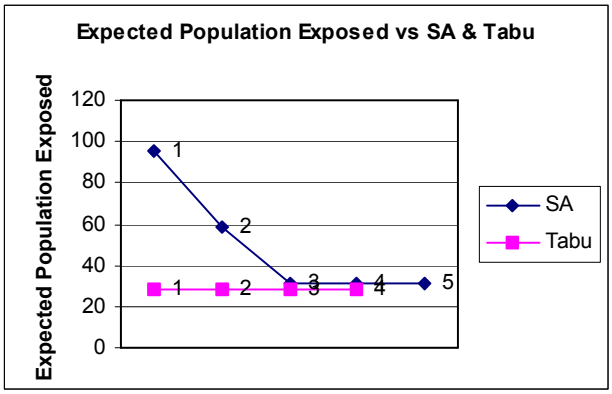


Figure 4.7 EPE for Randomized Neighborhood without Clustering

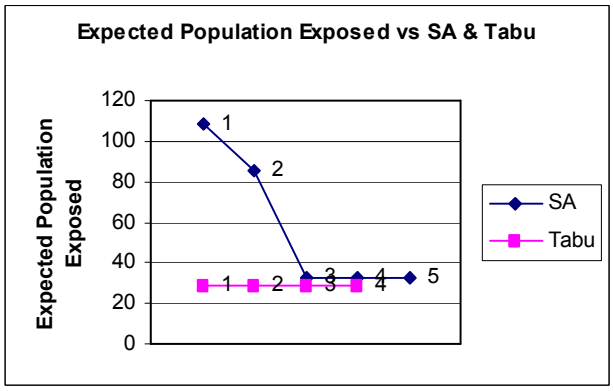


Figure 4.8 EPE for Randomized Neighborhood with Clustering

For the expected population exposed tabu provide a better solution than simulated annealing. From the depicted figures, the length of tabu size does not influence the solution.

3. Experiment in the Structured Neighborhood Search, Time vs Performance in Tabu

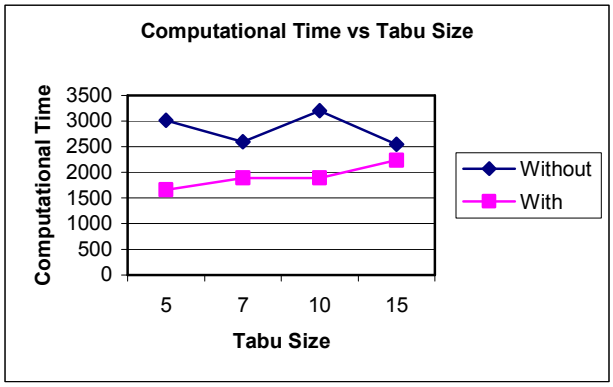


Figure 4.9 Computational Time vs Tabu Size ETD

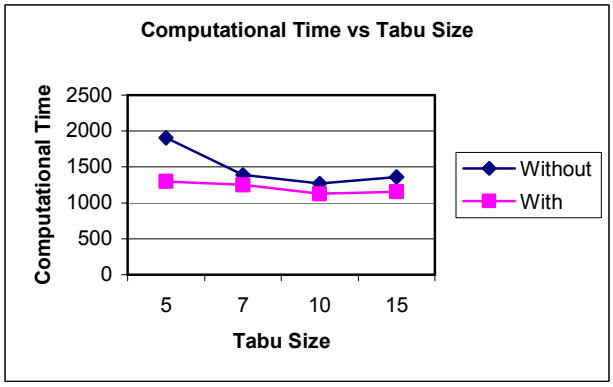


Figure 4.10 Computational Time vs Tabu Size EPE

As the figures illustrate that clustering in tabu search with structured neighborhood is superior to without clustering case with respect to computational time.

4. Experiment for the Randomized Neighborhood Search vs Time in Tabu

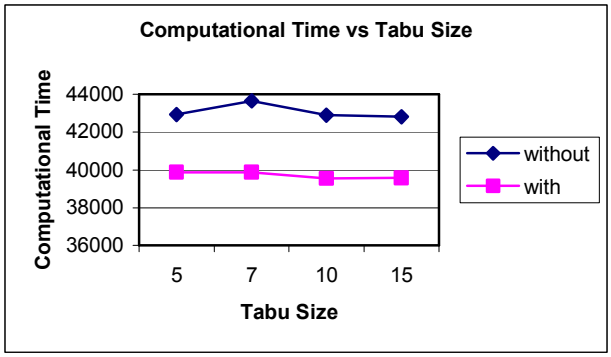


Figure 4.11 Computational Time vs Tabu Size ETD

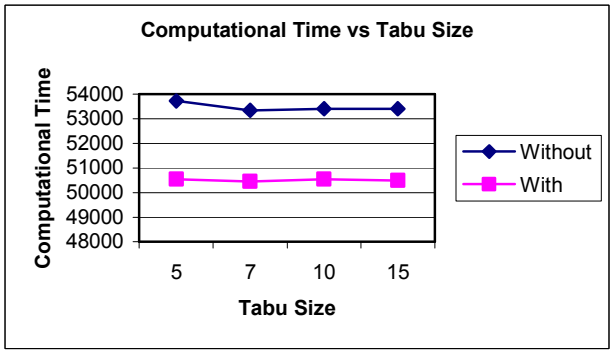


Figure 4.12 Computational Time vs Tabu Size EPE

Obviously, tabu search with randomized neighborhood search with clustering is more rapid than the case for without clustering. On the other hand, if we analyze the randomized versus structured neighborhood search with respect to their times, structured one depicts the fewer times for both of the clustering case.

5. Large network (1129 nodes) with 510 sensors located throughout the network is utilized for demonstrating the time versus performance with simulated annealing algorithm. Since we couldn't realize the exact difference for the solution with simulated annealing in moderate sized network with 92 nodes. We have illustrated the time versus performance relation for the structured neighborhood search.

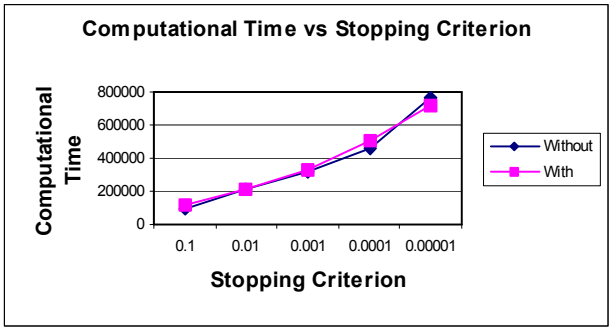


Figure 4.13 Computational Time vs SA Stopping Criteria ETD

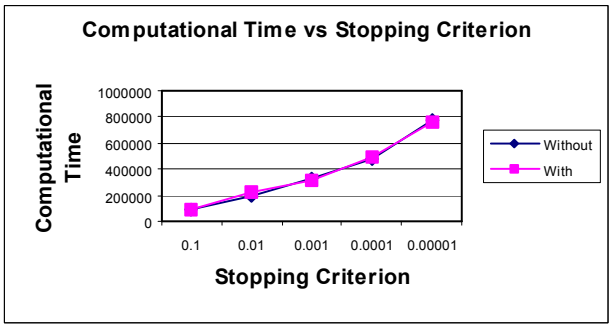


Figure 4.14 Computational Time vs SA Stopping Criteria EPE

There is no superior difference with respect to computational time in simulated annealing with structured neighborhood search according to expected time to detect and expected population exposed with and without clustering.

6. 1129 nodes with 510 sensors with Randomized Neighborhood Search in simulated annealing corresponding to a time versus performance figures.

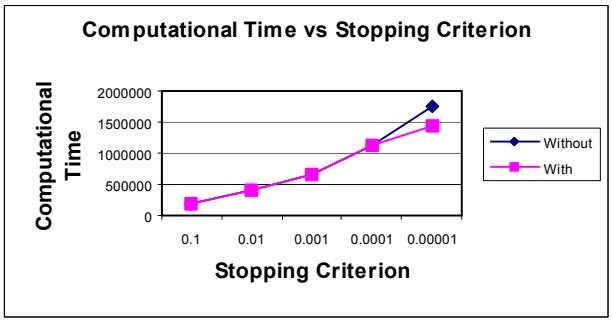


Figure 4.15 Computational Time vs SA Stopping Criteria ETD

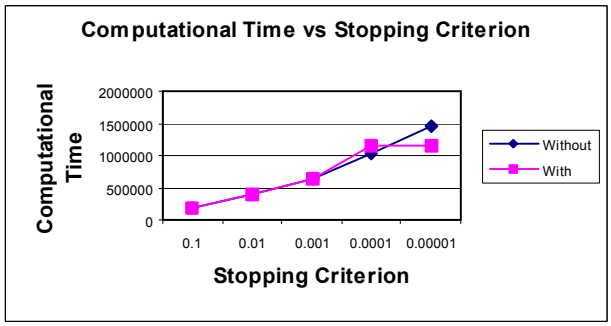


Figure 4.16 Computational Time vs SA Stopping Criteria EPE

As in the previous case, we could not say that with clustered or without clustered case improve the results. As in the both cases simulated annealing gives no difference between clustered or no clustered with respect to computational time.

On the other hand, it is easy to show that structured neighborhood search for the simulated annealing is relatively shorter computational time than the randomized neighborhood search.

7. Below figures depict the iterations number with respect to the expected time to detect and expected population exposed objective values in Simulated Annealing with stopping criterion 5 which is less than or equal to 10^{-5}. The experimentation is made with large network 1129 nodes with 422 sensors.

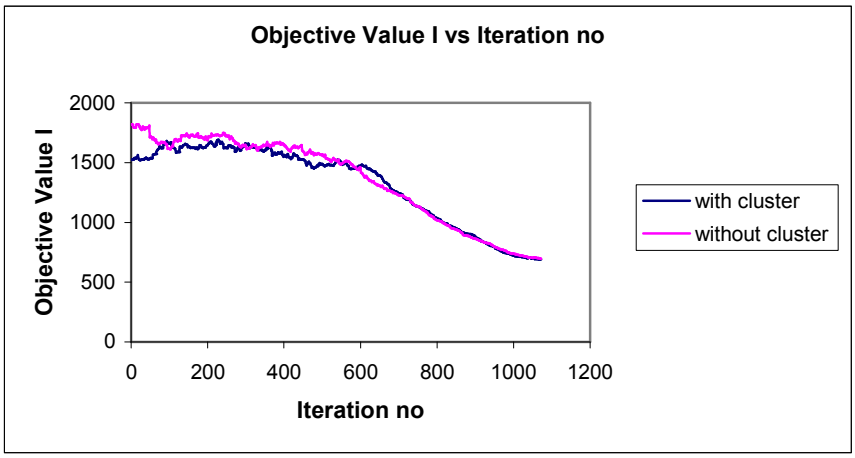


Figure 4.17 Expected Time to Detect vs Iteration no

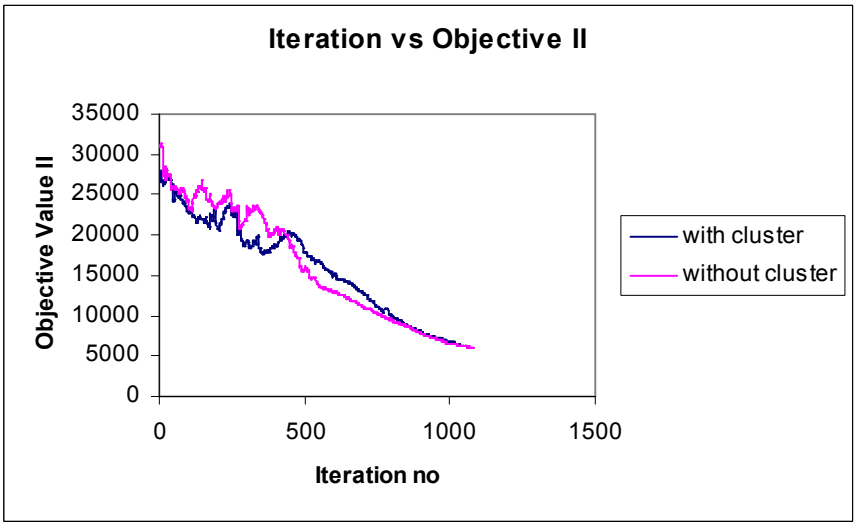


Figure 4.18 Expected Population Exposed vs Iteration no

From the figures, we can conclude that clustering gives better solution with smaller iteration numbers. As the iteration numbers increases, both cases (with or without clustering) present near solutions.

5 CONCLUSION

In this chapter, we provide a brief summary of the contributions done in this thesis and address of some of the future research directions. In this thesis, we have studied optimal sensors location in water distribution networks and concentrated mainly on two objectives: expected time to detect and expected population exposed. We proposed a new methodology to solve the binary nonlinear objectives. The solution procedures that are proposed are a meta-heuristics: simulated annealing and tabu search based on the two neighboring generation.

5.1 Contributions

First of all, we have introduced a novel approach so that the different flow patterns during a day can be cope with.

Secondly, we have considered an important feature of the problem domain that is not considered by the algorithms in the literature. That is the restricted sensor concept. In order to guarantee the full protection in the network, i.e. detection probability of an attack is equal 1; restricted sensors should be placed at the isolated node in the network.

Thirdly, in the proposed framework, we have realized the relationship among clustering and location problems in a way that the initial solutions of the meta-heuristics are produced with a k-medoids clustering algorithm.

Then we have presented the performance of the proposed algorithms that are affected by size of the networks and the number of sensors to be placed. Evidently, we showed that as the size of the network increases both the effect of clustering and the temperature level parameter becomes more significant especially in the large networks.

We also explored the competency of the algorithms with respect to some variations in the methodology with respect to the neighborhood generation mechanism: structured or randomized and selecting the initial solution based on clustering or not.

We also examined three different density levels: low, medium, and high in terms of the number of sensors that would be located.

We have accomplished that structured neighborhood search depicts the fewer computational times for with or without clustering case than the randomized one.

.

5.2 Future Research directions

There are several future research directions originated from this research study as follows:

1. In this research, we assumed that there is an equal probability of being a source of pollution node. This assumption may be relaxed by performing the unequal node probabilities for becoming a pollution node may be explored.
2. It is necessary to examine the multiple contaminant injection case.
3. In the case of sensors failure, no analysis has been made.
4. In this study, intrusion will occur only at nodes. In a more realistic case, contamination injections may be considered both along the pipe (edge) and at nodes. Hence, sensor placement at edges as well as at nodes could be examined.

6 REFERENCES

[1] Ahuja, R.K., Magnanti, T.L., and Orlin, J.B., Network Flows: Theory, Algorithms, and Applications, Prentice Hall Publisher, 147-148, 1993.

[2] Al-Zahrani, M. A. and Moied , K., 'Locating optimum water quality monitoring stations in water distribution system, Bridging the gap: Meeting the world's water and environmental resources challenges, State of the Practice Proc., World Water and Environmental Resources Congress, 2001.

[3] Berry, J., Fleischer, L., Hart, W.E. and Phillips, C.A., Sensor placement in municipal water networks, In Proceedings of the World Water and Environmental Resources Conference, 2003.

[4] Berry, J., Hart, W.E., Phillips, C.A., and Uber, J., A general integer-programming-based framework for sensor placement in municipal water networks, In Proceedings of the World Water and Environmental Resources Conference, 2004.

[5] Berry, J., Hart, W.E., Phillips, C.A., and Watson, J.P., Scalability of integer programming computations for sensor placement in water networks, ASCE Conference, 2005.

[6] Boccelli, D.L., Tryby, M.E., Uber, J., Rossman, L.A, Zierolf, M.L., and Polycarpou, M.M., Optimal scheduling of booster disinfection in water distribution systems, Journal of Water Resources Planning and Management, 124(2), 99–111,1998.

[7] Carr, R., Greenberg, H.J., Hart, W.E., and Phillips, C.A., Addressing modeling uncertainties in sensor placement for community water systems, In Proceedings of the World Water and Environmental Resources Conference, 2004.

[8] Geldreich, E.E., Investigating the outbreak in Cabool, Missouri for a water supply connection, Proceedings, AWWARF/EPA Conference on Water Quality Modeling in Distribution Systems, Environmental Protection Agency, Washington, D.C., 55-56, 1991.

[9] Ginsberg, M.D., and Hock, V.F., Terrorism and Security of Water Distribution Systems: A Primer, Defense and Security Analysis 20(4), 373-380, 2004.

[10] Glover, F., Tabu Search: Tutorial, Interfaces 20(4), 74-94, 1990.

[11] Harmant, P., Nace, A., Kiene, L., and Fotoohi, F., Optimal supervision of drinking water distribution network, Proc., 26th Annual Water Resources Planning and Management Conference, 1999.

[12] Hasan, J., States, S., Deininger, R., Safeguarding the security of public water supplies using early warning systems: A brief review, Journal of Contemporary Water Research and Education, 129, 27-33, 2004.

[13] Hromkovic, J., Algorithms for Hard Problems, Springer publisher, 389-400,2001.

[14] Kamber, M., Data Mining: Concepts and Techniques, Department of computer Science, University of Illinois at Urbana-Champaign, Chapter 8, 2004.

[15] Kumar, A., Kansal, M.L., and Arora, G., Identification of monitoring stations in water distribution systems, Journal of Environmental Engineering, 123(8), 1997.

[16] Kumar, A., Kansal, M.L., and Arora, G., Discussion of Detecting accidental contamination in municipal water networks, Journal of Water Resources Planning and Management, 124(4), 308-310, 1999.

[17] Laird, C.D., Biegler, L.T., Waanders, B.G, and Bartlett, R.A., Contamination source determination for water networks, Journal of Water Resources Planning and Management, 131(2), 125-134, 2005.

[18] Lee, B.H., and Deininger, R.A., Optimal locations of monitoring stations in water distribution system, Journal of Environmental Engineering, 118(1), 4–16, 1992.

[19] Mays, L.W., Water Supply Systems Security, McGraw-Hill publisher, Chapter 4-16, 2004.

[20] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A. H., and Teller, E., J. Chem. Phys. 21, 1087 (1953).

[21] Ostfeld, A., Kessler, A., and Sinai, G., Detecting accidental contaminations in municipal water networks, Journal of Water Resources Planning and Management, 124(4), 192-198, 1998.

[22] Ostfeld, A., Kessler, A., and Goldberg, I., A contamination detection system for early warning in water distribution networks, Engineering Optimization, 36(5), 525-538, 2004.

[23] Ostfeld, A., and Salomons, E., Optimal layout of early warning detection stations for water distribution systems security, Journal of Water Resources Planning and Management, 130(5), 377-385, 2004.

[24] Pool, S., and Lansey, K., Locating and operating disinfection boosters in water networks, Aesthetics in the constructed environment, Proc., 24th Annual Water Resources Planning and Management Conf., ASCE, New York, 738–743, 1997.

[25] Pool, S., Optimal operation and location of chlorine boosters, Master's thesis, University of Arizona, Tucson, Arizona, 2002.

[26] Propato, M., Piller, O., and Uber, J., A sensor location model to detect contaminations in water distribution networks, ASCE, 2005.

[27] R project for Statistical Computing, www.r-project.org

[28] Rossman, L.A., EPANET user's manual, Risk Reduction Engineering Lab., U.S. Environmental Protection Agency, Cincinnati, Ohio, 1994.

[29] Simha, R., Cai, W., and Spitkovsky, W., Simulated N-Body: New particle physics-based heuristics for a euclidean location-allocation problem, Journal Heuristics 7(1), 23-36, 2001.

[30] Thesen, A., Design and evaluation of tabu search algorithms for multiprocessor scheduling, Kluwer Academic Publishers, Madison USA, 1998.

[31] Tryby, M.E., and Uber, J., Representative water quality sampling in water distribution systems, Bridging the gap: Meeting the world's water and environmental resources challenges, State of the Practice—Proc., World Water and Environmental Resources Congress, 2001.

[32] Uber, J., Janke, R., Murray, R., and Meyer, P., Greedy heuristic methods for locating water quality sensors in distribution systems, ASCE Conference, 2004.

[33] Watson, J.P., Greenberg, H.J., and Hart, W.E., A multiple-objective analysis of sensor placement optimization in water networks, In Proceedings of the World Water and Environmental Resources Conference, ACSE, 2004.

[34] Woo, H.M., Yoon, J.H., and Choi, D.Y., Optimal monitoring sites based on water quality and quantity in water distribution systems, Bridging the gap: Meeting the world's water and environmental resources challenges, State of the Practice—Proc., World Water and Environmental Resources Congress, 2001.