

Optimal Sensor Locations in Water Distribution Networks

M. Farahian¹; M. J. Khanjani²; M. J. Fadaee³

1,2,3- Shahid Bahonar University, Kerman, Iran, Civil Engineering department.

Email: masoud_f_3t@yahoo.com

Abstract

Water distribution networks are inherently vulnerable to accidental or intentional water contaminations. There is a research aiming to prevent such threats in water networks. One of the problems arising is to identify sensor locations that minimize the impacts like contaminated volume consumed, population exposure and detection time of entrance of contamination. In this paper, we present a model of placing sensors in water network. we focus on the sensor configuration that minimizes the expected population exposed and time to detect. We formulate these problems as binary nonlinear objectives, and propose a solution based on simulated annealing. We simulated the contamination transport in EPANET in order to derive flow and velocity information. A novel approach also is introduced in order to handle different flow patterns occurring during the day.

Keywords: sensor placement, simulated annealing, expected population, optimization.

Introduction

Public water distribution systems are inherently vulnerable to accidental or intentional water contamination because of their distributed geography. Major accidental contamination events, like the defining accident on the River Rhine in Germany, have highlighted these vulnerabilities. Research on the problem of where to place sensors in water distribution networks to minimize the damage incurred by the intentional injection of chemical and biological contaminants has intensified since the attacks of the September 11, 2001; the vulnerability of these systems has become more widely appreciated. Accurate models of the sensor placement optimization problem now exist, in addition to methodologies for generating the associated model parameters. However, while algorithms for generating solutions to these models have been introduced, their applicability to large-scale, real-world water distribution networks is far from clear. First, most algorithms are tested exclusively on small scale networks, leaving open the question of scalability. Second, many algorithms are heuristic in nature and no effort has been made to establish empirical or theoretical performance bounds. Third, the modeling assumptions underlying some algorithms are physically unrealistic, raising questions regarding the utility of the resulting solutions in operational settings.

In this paper, we are aiming to propose a methodology, which might be utilized in order to determine sensor locations in the water distribution networks. We consider two objective functions; namely, the expected time to detect and expected population exposed. We propose a technique based on local heuristics, simulated annealing, and test its performance in various experimental conditions. Other than the solution methodology, the contributions of this study are: 1) to introduce 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. The rest of this paper is organized as follows. We briefly summarize and categorize prior research on sensor placement optimization in following part. Next, we describe our formulation of the sensor placement problem and this problem is discussed. Then, the modeling methodology is detailed. in the same section, we introduce the proposed algorithms to solve the problem. Next part is dedicated for the experimental results and discussion of the major findings.

¹ A student of MS Degree, Civil Engineering Department, Shahid Bahonar University, Kerman, Iran.

² Professor of Civil Engineering Department, Shahid Bahonar University, Kerman, Iran. P.o.box: 76135-666

³ Assistant Professor of Civil Engineering Department, Shahid Bahonar University, Kerman, Iran.

Optimization for Sensor Placement: Background

Conceptually, the objective in a sensor placement optimization problem (SPOP) is simple: to place a limited number of sensors in a water distribution network such that the impact to public health due to the accidental or intentional injection of contaminant is minimized. The broader research community has yet to arrive at a more specific, concrete definition that is widely (or even narrowly) agreed upon; research typically differs in terms of the precise definition of public health impact, the assumed characteristics of the deployed sensors, the fidelity of the contaminant transport simulation, and a host of other details. However, existing formulations of the SPOP can be usefully delineated in terms of the fidelity with which water quality changes resulting from an injection is captured. Two broad categories in the current literature can be identified, which we refer to simply as static and dynamic. In a *static* formulation of the SPOP, the impact of an attack at a particular network junction is estimated by analyzing some combination of (1) flow directions and velocities obtained via hydraulic simulation, (2) pipe lengths, and (3) junction demands. A prominent example of a static SPOP formulation is described by Kessler et al. [1998], and is based on the notion of an auxiliary network. An auxiliary network is a directed graph $G = (V, E)$ where elements of the set V represent nodes, e.g., junctions and tanks, of the distribution network. The edge set E is determined via analysis of hydraulic simulation outputs. For each pair of nodes v_i and v_j for which there is flow from v_i to v_j at any point in the simulation, a directed edge $e = (v_i, v_j)$ is added to E . Edges in $e = (v_i, v_j) \in E$ are weighted by the average velocity from v_i to v_j over the course of the simulation. The auxiliary graph is used in conjunction with the network pipe lengths to compute the shortest travel time between all pairs of vertices in the network. The travel times are then used to estimate the network-wide impact of an attack α at v_i if α is first detected by a sensor located at a vertex v_j ; the specific measure of health impact considered by Kessler et al. is the total volume of contaminated water consumed before detection by at least one sensor. Kessler et al. solve their static formulation of the SPOP via heuristic solution of a corresponding set cover problem. Berry et al. [2003] introduce a static SPOP formulation in which the objective is to minimize the expected fraction of the population exposed to an injected contaminant. Here, hydraulic simulation results are used to compute a fixed flow orientation for each pipe in the network over a series of p distinct non-overlapping time intervals, referred to as patterns. The formulation is time-independent, in that travel times are not considered; rather, a node v_j is protected against an attack at vertex v_i if and only if there is a sensor capable of detecting the flow between v_i and v_j . Watson et al. [2004] generalize the Berry et al. formulation to consider a range of optimization objectives, some of which account for travel times in a manner consistent with that of the Kessler et al. formulation. Both Berry et al. and Watson et al. solve the resulting SPOP formulations via exact solution of corresponding mixed-integer programs.

There are two key assumptions underlying any static SPOP formulation, e.g., that of Kessler et al. First, factors such as contaminant dilution, concentration level, and mode of attack are not modeled. Rather, the static SPOP simply tracks the *projected* presence or absence of contaminant at various network points over time, and assumes identical contaminant and water flow dynamics. Second, the contaminant transport model is based on aggregated flow velocities, such that the true dynamics of the underlying flow are only approximated. Each of these assumptions represents a potentially significant deviation from reality, and the impact of these approximations on the quality of solutions to the SPOP are currently poorly understood.

In contrast, *dynamic* SPOP formulations correct for each of the aforementioned deficiencies by precisely characterizing the impact of an attack at a given network junction on the rest of the network. First introduced by Ostfeld and Salomons [2004], dynamic SPOP formulations use detailed water quality simulation results to compute contaminant concentration time-series for each junction in the network. These time-series can be used to determine the impact of an attack α at v_i if α is first detected by a sensor located at a vertex v_j . In addition to accuracy improvements relative to their static counterparts, dynamic SPOP formulations have the added advantage that a full range of attack types and sensor characteristics can be modeled, as the network response is completely specified by contaminant level time-series at each network junction; the accuracy of the formulation is strictly limited by the accuracy of the water quality simulation.

Mirroring the earlier approach of Kessler et al., Ostfeld and Salomons solve their dynamic SPOP via solution of a corresponding set covering problem. However, the optimization objective is more realistic: to minimize the volume of polluted water possessing a concentration of pollutant higher than a minimum hazard level. Most recently, Berry et al. [2004] discuss a dynamic SPOP formulation for minimizing the expected volume of contaminated water consumed before detection, which is expressed and solved as a mixed-integer program. Finally, we observe that the accuracy of dynamic SPOP formulations comes with a price, specifically in the form of a very large number of

computationally intensive water quality simulations; in contrast, static SPOP formulations are based strictly on comparatively cheap hydraulic simulations.

Sensor Placement; Problem designing method

We now introduce the specific formulation of the dynamic SPOP used in our analysis. Our objective is to minimize the total volume of contaminated water consumed, at any concentration level and expected time to detect the contamination. We have two distinct problems. First, we assume that we have a budget of p of sensors that can be placed at any junction in a distribution network, each sensor is capable of detecting any concentration level of contaminant, and a general alarm is immediately raised when contaminant is detected such that all further consumption is prevented. As discussed in part 2, we observe that none of these assumptions are binding, and can be relaxed without impacting the mathematical structure of our formulation. We view the structure of a water distribution network as an undirected graph $S = (V, E)$; elements of the set V represent junctions and sources, while elements of the set E represent pipes, pumps, and valves.

Let A denote the set of attacks against which a sensor configuration consisting of p sensors is intended to protect. We assume attacks can occur at any vertex $v \in V$ of the network, i.e, injection via backflow is possible. Elements of $a \in A$ are quadruples of the form $a = (v_x, t_s, t_f, X)$, where $v_x \in V$ is the attack vertex, t_s and t_f are the attack start and stop times, and X is the attack profile (e.g., arsenic injected at a particular concentration at a given rate). For each attack $a \in A$, we use existing water quality analysis software (e.g., as found in EPANET [Rossman, 1999]) to compute the contaminant concentration at each node in the network from time t_s to an arbitrary point $t_h \geq t_f$ in the future. The results of such an analysis are expressed in terms of concentration time-series τ_j for each $v_j \in V$, with samples at regular (arbitrarily small) intervals within $[t_s, t_h]$. Using the set of τ_j in conjunction with demand profiles, it is straightforward to compute the total volume of contaminated water $d_{a(t)}$ consumed (network-wide) due to an attack a at any given point at time $t \in [t_s, t_h]$. Next, let a_j denote the earliest time t at which a hypothetical sensor at vertex v_j can detect contaminant due to an attack a ; $a_j = t_h$ if no contaminant ever reaches v_j , and a_j can be easily computed from τ_j . Finally, we define $d_{aj} = d_i(\gamma_{aj})$, i.e., the total volume of contaminant consumed due to an attack a if the attack is first detected by a sensor at v_j . Given a set A of attack scenarios, a set V of network vertices, and a set d_{aj} of impact parameters, we take as our design objective like Watson et al. [2004] the minimization of the aggregate impact I over all attack scenarios, where

$$I = \sum_{i=a}^{|A|} \sum_{j=1}^{|V|} d_{aj} x_{aj} \quad (1)$$

Subject to the constraints:

$$\sum_{j=1}^{|V|} x_{aj} = 1, \quad \forall a \in A \quad (2)$$

$$x_{aj} \leq s_j, \quad \forall a \in A, \forall j \in V \quad (3)$$

$$\sum_{j=1}^{|V|} s_j = p \quad (4)$$

$$0 \leq x_{aj} \leq 1, \quad s_j \in \{0,1\}, \quad \forall a \in A, \forall j \in V \quad (5)$$

A variable s_j (Constraint 5) indicates whether one of the p available sensors is placed on vertex v_j , while Constraint 4 requires that a total of exactly p sensors be placed. A variable x_{aj} (Constraint 5) indicates whether an attack $a \in A$ is detected by a sensor at vertex v_j ; Constraint 3 enforces the condition that detection can only occur at v_j if a sensor is placed there. Finally, Constraint 2 requires that detection of each attack $a \in A$ be assigned to a single vertex v_j ; in other words, there is always a first vertex in the network to detect an attack. We observe that this formulation is conceptually identical to the dynamic SPOP introduced by Berry et al. [2004]. Our variant is more explicit, for reasons discussed below, in that d_{aj} are defined for all possible combinations of attack $a \in A$ and vertex $v_j \in V$ – despite the fact that in practice it is typically not possible for contaminant to flow between arbitrary a and v_j .

Now, we introduce the second objective function; the minimization of the expected time to detect:

$$D = 1/V \left(\sum_{j=1}^{|V|} C_j \right) \quad (6)$$

Subject to the constraints:

$$C_i = \min_j \{d_{ij}; x_j = 1\} \quad (7)$$

$$\sum_{j=1}^v x_j = p \quad (8)$$

$$x_j \in \{0,1\} \quad (9)$$

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 unites between the attack node j , v 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 formation, the objective function corresponds to minimizing the expected time to detect. Constraint (1) gives the minimum distance if the sensor located at 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 .

design methodology

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

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 evening 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. [1998] proposed to work with three periods of 8-hours and different demand patterns associated with each time period at each node. 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. [1998] 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 combination. 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 unites. 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. If pipe length is assumed 1000 unites, 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 results are depicted in Figure1.

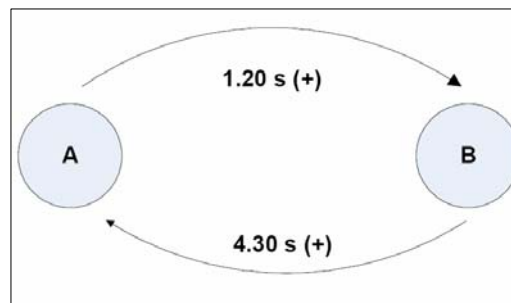


Figure 1- Representative flow pattern

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. [1998] 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 unidirectional and parallel pipes. Moreover, Ostfeld et al. [1998] 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 directions of the flow changes from node B to node A after 4 PM. Ostfeld's et al. [1998] approach assumes that there is a certain (average) flow at both directions throughout the day. Based on this assumption a contaminant intrusion 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 can not be detected until the direction of the flow changes. Therefore, the calculations based on the averaging technique of Ostfeld et al. [1998] are misleading and should be corrected. In this study, calculation of the flow patterns is different than the calculations in Ostfeld et al. [1998]. 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 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 2.

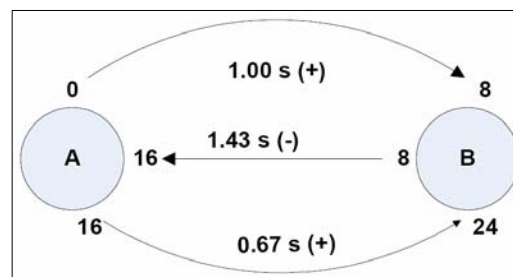


Figure 2- proposed flow pattern

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. All data about demand pattern, pipe length, pipe diameter, tank elevation, reservoir and pump are basic needs for simulation of the network with EPANET.

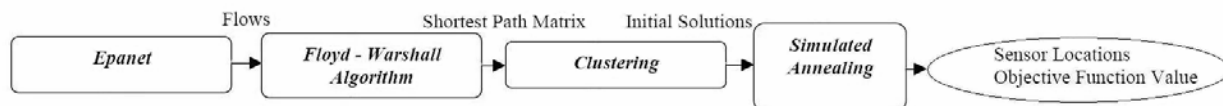


Figure 3- The flow of the Methodology

Simulated annealing

Metropolis et al. [1953] first introduced Simulated annealing (SA) also known as Monte Carlo annealing in 1953. Simulated annealing (SA) also known as 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.

Simulated annealing is a widely recognized optimization tool. according to Juraj Hromkovic [2001] SA is a 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 simulation algorithm.

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 , $0.8 \leq r \leq 0.99$. using this reduction one works a constant number k of iterative steps with a fixed T , and after k steps $T = r * T$.

In our research, 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 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;

$$p = e^{-\Delta E/T} \quad (10)$$

where t represents the actual temperature and ΔE represents the difference between the fitness value of the candidate solution and 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.

In this paper, 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. 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, 0.0001, 0.00001 and 0.000001. 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.

Computational results and experimental analysis

Now, 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 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.

There are two factors that can affect the efficiency of the proposed algorithms: size of the network and number of sensors located. In this paper we utilized three water distribution networks. The first one is a small network with only 9 nodes. The second is a network with 33 nodes and third with 92 (table 1). We also considered three different density levels in terms of the number of sensors that would be located. 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

Table 1- size of the network and density category of sensors

Density of nodes	9	33	92
Low density	2	7	19
Medium density	3	10	27
High density	4	13	35

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 metaheuristics. In order to make this paper shorter, we did not include all the details.

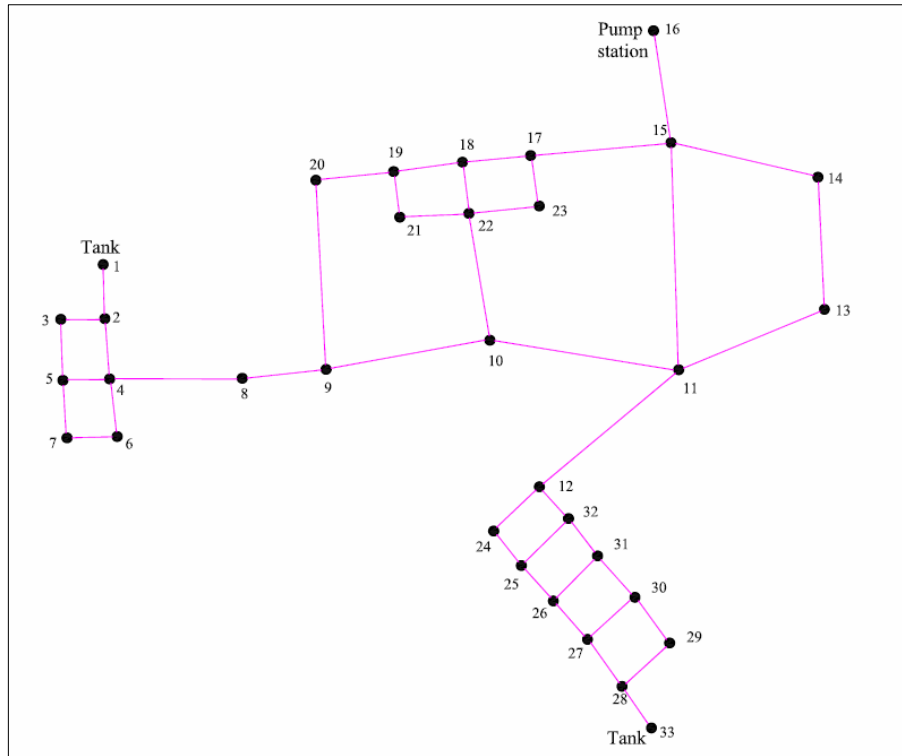


Figure 4- view of second network with 33 nodes

Results

In table 2, the expected time to detect values are presented for SA algorithm with respect to structured neighborhood search. In the same table, the objective function values corresponding to different temperature level parameters also presented. It seems that as size of the network increases the effect of temperature level parameter becomes more significant.

Table 2- Expected Time to Detect with Structured Neighborhood Search for SA

nodes	density	Simulated annealing stopping criteria				
		0.1	0.01	0.0001	0.00001	0.000001
9	2	3.94	3.94	3.94	3.94	3.94
	3	2.51	2.51	2.51	2.51	2.51
	4	1.62	1.62	1.62	1.62	1.62
33	7	8.23	7.64	7.64	7.64	7.64
	10	5.44	4.31	4.31	4.31	4.31
	13	4.51	3.92	3.53	3.53	3.53
92	19	2.67	2.14	1.88	1.88	1.88
	27	1.61	1.34	1.00	1.00	1.00
	35	1.10	0.83	0.61	0.61	0.61

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. In Table 3, the expected population exposure values are presented for the SA

algorithm with respect to the structured neighborhood search. In the same table, the objective function values corresponding to different temperature level parameters also presented.

Table 3- Expected Population Exposed with Structured Neighborhood Search for SA

nodes	density	Simulated annealing stopping criteria				
		0.1	0.01	0.0001	0.00001	0.000001
9	2	958.15	958.15	958.15	958.15	958.15
	3	400.25	400.25	400.25	400.25	400.25
	4	192.96	192.96	192.96	192.96	192.96
33	7	174.66	164.42	164.42	164.42	164.42
	10	97.90	78.62	78.62	78.62	78.62
	13	61.39	38.04	33.58	33.58	33.58
92	19	479.76	350.32	118.73	118.73	118.73
	27	163.81	102.73	42.29	37.90	37.90
	35	79.06	59.95	31.86	28.47	28.47

Conclusion

in this paper, we have studied optimal sensor locations 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 procedure that proposed are simulated annealing based on structured neighborhood generation. First, we have introduced a novel approach so that the different flow patterns during the day can be cope with. Then we have considered an important feature of the problem domain that is not considered by the algorithms in the literature. 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. We 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, the effect of temperature level parameter becomes more significant.

References

1. A. Kessler, A. Ostfeld, and G. Sinai. Detecting accidental contaminations in municipal water networks. *Journal of Water Resources Planning and Management*, 124(4): 192–198, July/August 1998.
2. J. Berry, L. Fleischer, W.E. Hart, and C.A. Phillips. Sensor placement in municipal water networks. In *Proceedings of the World Water and Environmental Resources Conference*, 2003.
3. J.P. Watson, H.J. Greenberg, and W.E. Hart. A multiple-objective analysis of sensor placement optimization in water networks. In *Proceedings of the World Water and Environmental Resources Conference*. ASCE, 2004.
4. A. Ostfeld and E. Salomons. Optimal layout of early warning detection stations for water distribution systems security. *Journal of Water Resources Planning and Management*, 130(5):377–385, 2004.
5. J. Berry, W.E. Hart, C.A. Phillips, and J. Uber. A general integer-programming-based framework for sensor placement in municipal water networks. In *Proceedings of the World Water and Environmental Resources Conference*, 2004.
6. L.A. Rossman. The EPANET programmer’s toolkit for analysis of water distribution systems. In *Proc. of the Annual Water Resources Planning and Management Conference*, 1999.
7. Ostfeld, A., Kessler, A., and Sinai, G., Detection accidental contaminations in municipal water networks, *Journal of water resource planning and management*, 124(4), 192-198, 1998
8. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A. H., and Teller, E., *J. Chem. Phys.* 21, (1087), 1953
9. Hromkovic, J., *Algorithms for Hard Problems*, Springer publisher, 389-400,2001.