Simultaneous Sensor and Actuator Placement for Contaminant Detection and Mitigation in Water Distribution Networks

Suhas Gundimeda[a], Venkata Reddy Palleti[a], Shankar Narasimhan[a]

[a]*Department of Chemical Engineering, Indian Institute of Technology Madras, Chennai 600036, India*

Abstract

Water Distribution Networks (WDNs) are considered to be part of the critical infrastructure of any city. WDNs are often exposed to either accidental or intentional contamination. Therefore, it is very important to protect the public form such kinds of incidents. These contaminants can be detected by deploying sensors in the work. Once the sensors detects the presence of a contamination, it is also necessary to take appropriate actions in order to mitigate the effects of contaminations. Therefore, in this paper, we propose an optimal simultaneous sensor and actuator placement for contamination detection and mitigation in WDNs. A multi-objective mixed integer linear optimization formulation is proposed to obtain the optimal sensor and actuator placement. The proposed method is demonstrated on the real urban water distribution system.

Keywords: Water distribution networks, observable sensor network, actuator network, contamination detection and mitigation.

1. Introduction

Water Distribution Networks (WDNs) are networks of pipes, tanks, pumps, valves, and other components that are used to distribute water from sources, like reservoirs, to consumers. The topology of the network, number of consumer nodes, loading conditions, hydraulic elements etc make the control of a water distribution system a difficult task and makes it highly vulnerable to contamination (?). WDNs are often susceptible to either accidental or intentional contamination. Accidental contamination occurs due to system deficiencies (e.g, cross connections, pipeline breakage) and intentional contamination occurs as a result of deliberate acts of chemical or biological intrusions. Introduction of chemical or biological agents in WDN can damage public health. After 9/11 attacks in United States, the concern over possible terrorist attacks on WDN has increased and can be considered as the most serious threat to WDNs (?). Hence, it is very important to detect such kinds of attacks as quickly as possible and it is also necessary to take appropriate actions to mitigate the effects of a contamination event. Suggested detection and response methods are varied, and one such is the placing of online sensors and actuators (or shut-off valves) for detection and containment of contaminants. The sensors can be used to detect the contaminants, and the actuators are useful to shut down the network to prevent further spread of contaminants. Therefore, in this paper we propose a methodology to obtain an optimal sensor and actuator placement for contaminant detection and mitigation of the contamination effects.

The problems of contaminant detection in WDNs and appropriate response to a contamination event have been addressed in literature. Numerous procedures are available in the literature for the contamination detection problem. However, very few studies are available with respect to the problem of response to contamination events. Several optimization algorithms have been proposed for the problem of sensor placement for contamination detection in WDNs. Single objective optimization problems(????) are solved by considering objectives for maximizing the demand coverage, maximizing the likelihood detection and minimizing the elapsed time between the injection and detection of contaminants. Further, multi-objective optimization problems (????) have been solved by considering objectives for minimizing the time of detection, minimizing the contaminated volume consumed prior to detection, population at risk etc. The problem of contamination source identification based on deployed sensor measurements have been reported (??). Sensor placement algorithm using the principle that there must exist a unique non-zero set of sensors for each of the vulnerable nodes was developed by Palleti et al. (2016). In this paper, the graph abstraction of water distribution networks with vulnerable nodes representing multiple real nodes and used sets of affected nodes to find a viable sensor placement.

Although, design of a sensor network to detect and identify intentional contamination of a WDN is important, it is also necessary to consider the potential action that one can take to mitigate the effect of the contaminant. Once the sensor network detects the presence of a contaminant, U.S. Environmental Protection Agency's (USEPA) response protocol toolbox (?) provides recommendations for implementation of specific responses to minimize the effect of contamination event. This protocol includes the detection, source identification followed by consequence management strategies. Various researchers have developed methodologies related to the detection and source identification problems as described earlier. Once a contaminant has been detected, water utilities must evaluate the response to the contamination so that potential impact to the public can be minimized. These consequence management strategies may include: (1) public notification, (2) isolation and containment of a contaminant through valve operations (?), (3) flushing the contamination system, and (4) combinations of isolation, notification, and flushing (?). Very few studies are available with respect to the mitigating actions that would be taken after detecting the presence of contaminant in the WDN (?????) . Heuristic procedures are proposed for isolating a contaminated area through the simultaneous closure of a number of valves in the system, assuming an unlimited number of response teams and subsequent removal of the contaminant by unidirectional flushing ?? . ? proposed a multi-objective approach with the aim of minimizing the number of operations to be performed and to reduce the consumption of contaminated water by consumers for a given specific contamination event. Such an approach is useful only when complete knowledge of the contaminant history is known (in terms of intrusion location, duration and quantity of contaminant injected). Similarly, multi-objective procedures were proposed to identify the minimum number of operations that minimize the contaminated water consumption (??). These approaches (??) have not considered the characteristics of the contamination event (i.e, location, duration and quantity of contaminant introduced). Later, the problem of optimal scheduling of device activation in order to minimize the consumption of contaminated water after the detection of a contaminant has been studied by ?.

The previous studies focused on operational strategies to be implemented in the event of a contaminant detection assuming that valves are already present in the network. The design problem of optimal valve placement to minimize the effects of contamination events has not been reported in the literature. But recently, ? proposed a methodology for design of actuator network for a given sensor network to mitigate contamination effects in WDNs. In their work, shut-off valves are placed on optimal number of pipes so that it is possible to shutdown the network to prevent the uncontaminated part of the network from the contaminated network. Further, they design the actuator network assuming that the sensor network design has already been carried out and locations of sensors are known a priori. It means that the sensor and actuator network designs are carried out independently and they are decoupled. In contrast to this work, in this paper, we obtain a sensor and actuator placement simultaneously for detection and mitigation of contamination effects. Therefore, in this work we propose a multi-objective mixed integer linear optimization formulation is proposed to determine the simultaneous optimal sensor and actuator network. Also, we hypothesize that simultaneous sensor and actuator planning can be achieved and is more efficient than iterative non-linear optimization sequentially solving the two placement problems, i.e. these are not independent problems, and these can be formulated as an integer linear program and solved.

2. Problem Description

The water distribution network is represented as a deterministic weighted directed graph, $G(V, E)$, where E represents the edges or connections, and V represents the vertices or nodes. Nodes are used to represent sources, such as reservoirs or tanks from where water is supplied, as well as demand points where water is consumed and taken out of the network. All the junctions of pipes, such as those from pipes dividing or combining are also regarded as nodes. Pipes, valves, pumps, and other connections are represented as edges in the graph. A typical city's WDN has several hundred nodes and edges. Intentional contamination generally only occurs at sources - tanks, reservoirs, fire hydrants. Unintentional contamination can typically occur at any point in the WDN. In this work, we develop techniques that can be used for the general case, where any set of nodes in the network can be contaminated. These potential sites of intrusion are referred to as vulnerable nodes.

We are given complete information about the network - (sparse) adjacency matrix, list of nodes vulnerable to contamination, list of nodes which shouldn't be contaminated (demand nodes). All of the vulnerable nodes are infected (not necessarily at the same time) - case 1. A subset of vulnerable nodes are contaminated - this required a modified solution - case 2. There is no time delay in sensing, actuation, communication.

2.1. Requirements to be satisfied

To find distribution of sensors on nodes and actuators on edges such that an attack on a subset of vulnerable nodes can be identified and the network shut down, preventing the contaminant from reaching the demand nodes.

Secondary requirement A sensor and actuator placement strategy that might allow for initial contamination of some demand nodes, but will stop any further contamination to all demand nodes.

Formulating the problem as Mixed Integer Linear Program (MILP)

A binary integer optimization problem is presented:

$$
\begin{aligned}
\text{PROBLEM I: } & min(\sum x_i + \sum z_i) \\
\text{s.t. } & \mathbf{A}p \le \mathbf{b} \\
& \mathbf{A_{eq}}p = \mathbf{b_{eq}}
\end{aligned}
\tag{1}
$$

Where, $p = \begin{bmatrix} \mathbf{x} & \mathbf{y} & \mathbf{z} \end{bmatrix}^T$

$\mathbf{x} \equiv [x_i]$ is 1 if there exists a sensor at i^{th}node, 0 otherwise.

$\mathbf{y} \equiv [y_i]$ is 1 if i^{th}node is in the demands side of the actuators, 0 otherwise.

$\mathbf{z} \equiv [z_i]$ is 1 if there exists an actuator at i^{th}edge, 0 otherwise.

Hence the vector $\begin{bmatrix} \mathbf{x} & \mathbf{y} & \mathbf{z} \end{bmatrix}$ is of length $2 * N + E$.

$A = \begin{bmatrix} A_1 & \mathbf{0} \\ \mathbf{0} & A_2 \end{bmatrix}$, $A = \begin{bmatrix} A_{eq1} & \mathbf{0} \\ \mathbf{0} & A_{eq2} \end{bmatrix}$, $b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$, $b_{eq} = \begin{bmatrix} b_{eq1} \\ b_{eq2} \end{bmatrix}$.

A_1 is a matrix of vulnerable nodes vs. affected nodes, with $A_{1ij} = -1$ if i^{th} vulnerable node affects the j^{th} node in graph.

$A_2 = \begin{bmatrix} J & I_{E\times E} \end{bmatrix}$ where J is the incidence matrix.

$b_1 = \begin{bmatrix} . \\ . \\ . \\ -1 \\ . \\ . \\ . \end{bmatrix}$, $b_2 = [\mathbf{O}]$. This links the partitioning variables $\mathbf{y}$ to the variables in the cost function, $\mathbf{z}$.

$A_{eq1} = b_{eq1} = \mathbf{O}$

A_{eq2}, b_{eq2} provide bounds to force the decision variables to reality, i.e. the demand nodes being in the second partition and the vulnerable nodes being in the first.

Small network example

Consider the graph represented by Figure 1, with demand node 6 and vulnerable nodes 1 and 2.

Formulating as binary integer optimization problem:

$$\begin{array}{ccc} & min \quad (\sum x_i + \sum z_i) & \\ & sub & \\ \mathbf{A}\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} & \leq & \mathbf{b} \\ \mathbf{A_{eq}}\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} & = & \mathbf{b_{eq}} \end{array}$$

Hence the vector $\begin{bmatrix} \mathbf{x} & y & z \end{bmatrix}$ is of length $2 * N + E = 21$.

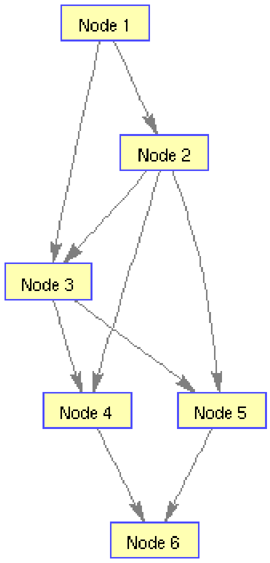

Figure 1: The example graph with 6 nodes and 9 edges

$$A = \begin{bmatrix} A_1 & \mathbf{0} \\ \mathbf{0} & A_2 \end{bmatrix}, A = \begin{bmatrix} A_{eq1} & \mathbf{0} \\ \mathbf{0} & A_{eq2} \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, b_{eq} = \begin{bmatrix} b_{eq1} \\ b_{eq2} \end{bmatrix}.$$

A_1 is a matrix of vulnerable nodes vs. affected nodes, with $A_{1ij} = -1$ if i^{th} vulnerable node affects the j^{th} node in graph.

$$A_1 = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

$$A_{eq1} = b_{eq1} = \mathbf{O}$$

The following maps the actuator placement problem to one of classification into source or demand partition.

$$A_2 = \begin{bmatrix} J & I_{E\times E} \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

where J is the incidence matrix.

$b_2 = [\mathbf{O}]$. This links the partitioning variables $\mathbf{y}$ to the variables in the cost function, $\mathbf{z}$. An edge between two different partitions causes the cost function to assume the presence of an edge.

A_{eq2}, b_{eq2} provide bounds to force the decision variables to reality, i.e. the demand nodes being in the second partition and the vulnerable nodes being in the first.

$$A_{eq2} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; b_{eq2} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Results and verification

It takes three sensors to detect contaminant with no other requirements.

This is the same as Palleti et al. (2016).

The closing of network requires 12 actuators: on edges 1, 4, 30, 37, 39, 47, 49, 50, 64, 81, 149, 154.

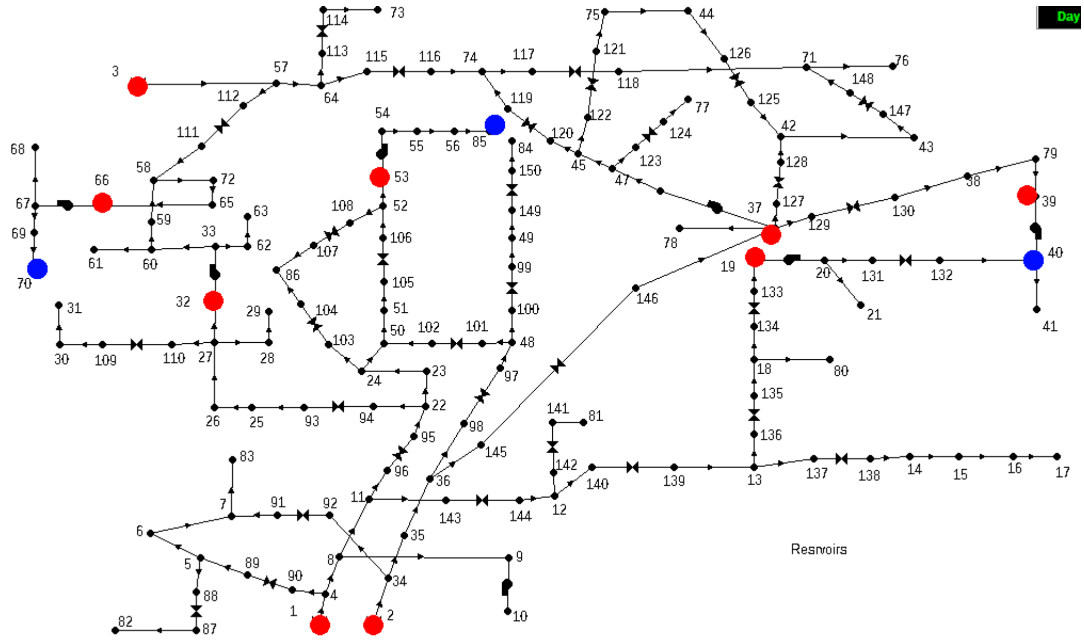

Figure 2: Graph abstraction with vulnerable nodes in red and sensors in blue.

Case 2: The contaminant must be contained in the vulnerable side of actuator network; attacks occur on all vulnerable nodes at once

This case is not trivial as the positions of the sensors must be used as input, i.e. they must be on the vulnerable side of actuators.

Formulation

Formulating as binary integer optimization problem:

Vulnerable Nodes	Sensor Nodes
1	41, 70, 85
2	41, 85
3	70
19	41
32	70
37	41
39	41
53	85
66	70

Table 1: Vulnerable nodes and their corresponding sensor nodes

$$min \quad (\sum x_i + \sum z_i)$$

$$sub$$

$$\mathbf{A}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\\ a\\ \mathbf{w}\end{bmatrix} \leq \mathbf{b}$$

$$\mathbf{A_{eq}}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\\ a\\ \mathbf{w}\end{bmatrix} = \mathbf{b_{eq}}$$

Where $\mathbf{x} \equiv [x_i]$ is 1 if there exists a sensor at i^{th}node, 0 otherwise.

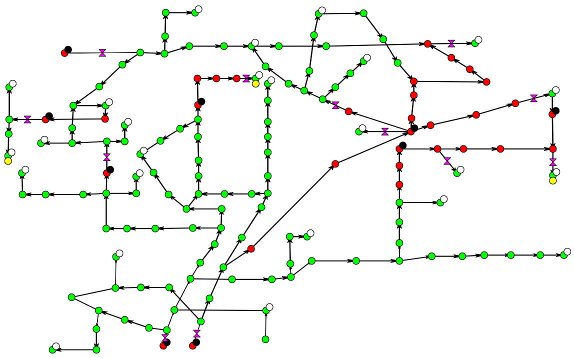

Figure 3: Results of independent placement case

$\mathbf{y} \equiv [y_i]$ is 1 if i^{th}node is in the demands side of the actuators, 0 otherwise.

$\mathbf{z} \equiv [z_i]$ is 1 if there exists an actuator at i^{th}edge, 0 otherwise.

a is a decision variable which stores the maximum distance the set of sensors are to the set of vulnerable nodes.

$\mathbf{w} \equiv [w_i]$ is 1 if i^{th}node is in the demands side of the actuators, a large number(N) otherwise.

M is a large number.

Hence the vector $\begin{bmatrix} \mathbf{x} & y & z & a & w \end{bmatrix}$ is of length $3 * N + E + 1$.

$$A = \begin{bmatrix} A_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & A_2 & \mathbf{0} \\ & A_3 & \end{bmatrix}, A = \begin{bmatrix} A_{eq1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & A_{eq2} & \mathbf{0} \\ \mathbf{0} & A_{eq3} & \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, b_{eq} = \begin{bmatrix} b_{eq1} \\ b_{eq2} \\ b_{eq3} \end{bmatrix}.$$

A_1 is a matrix of vulnerable nodes vs. affected nodes, with $A_{1ij} = -1$ if i^{th} vulnerable node affects the j^{th} node in graph.

$$b_1 = \begin{bmatrix} . \\ . \\ . \\ -1 \\ . \\ . \\ . \end{bmatrix}$$

$A_{eq1} = b_{eq1} = [\mathbf{O}]$

$A_2 = \begin{bmatrix} J & I_{E\times E} \end{bmatrix}$ where J is the incidence matrix.

$b_2 = [\mathbf{O}]$. This links the partitioning variables $\mathbf{y}$ to the variables in the cost function, $\mathbf{z}$.

A_{eq2}, b_{eq2} provide bounds to force the decision variables to reality, viz. the demand nodes being in the second partition and the vulnerable nodes being in the first.

We use shortest path lengths from all the vulnerable nodes (simulating an attack on all of them) to all the nodes in the graph to model "farther away in time". Let this vector be $\mathbf{S}$. The first set of constraints of A_3 is this vector of shortest paths multiplied with each column of $\mathbf{I} \times \mathbf{x}$, storing the maximum distance of that particular sensor configuration in a. The actuators need to be at least that much distance away from the set of vulnerable nodes.

This can be implemented in two ways: constraining the actuators to be placed after a particular distance, or constraining the demand partition to start after a particular distance. Since generally speaking $N < E$ in a lot of real world networks, we show the node implementation.

We transform the $\mathbf{y}$ vector to $\mathbf{w}$, where M denotes the sensor partition and 1 implies the demand partition. Then we construct constraints for actuator placement for the i^{th}node as follows: $\begin{bmatrix} ... & 0 & -(M+\mathbf{S}) & 0 & ... \end{bmatrix} \mathbf{w}_i + \begin{bmatrix} a + \frac{a}{M} \end{bmatrix} \leq \begin{bmatrix} -M \end{bmatrix}$. This completes the formulation.

Small network example

Consider the same network in Figure 1, with the two vulnerable nodes (1, 2) and one demand node (6).

The additional constraints introduced in this case are $\mathbf{S}_i\mathbf{x}_i - a \leq 0$ for each node i. This becomes

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix} \mathbf{x} - \begin{bmatrix} a \\ a \\ a \\ a \\ a \\ a \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The constraints for ensuring desired partitioning are:

$$\begin{bmatrix} -(M) & 0 & 0 & 0 & 0 & 0 \\ 0 & -(M) & 0 & 0 & 0 & 0 \\ 0 & 0 & -(M+1) & 0 & 0 & 0 \\ 0 & 0 & 0 & -(M+1) & 0 & 0 \\ 0 & 0 & 0 & 0 & -(M+1) & 0 \\ 0 & 0 & 0 & 0 & 0 & -(M+2) \end{bmatrix} \mathbf{w} + \begin{bmatrix} a+\frac{a}{M} \\ a+\frac{a}{M} \\ a+\frac{a}{M} \\ a+\frac{a}{M} \\ a+\frac{a}{M} \\ a+\frac{a}{M} \end{bmatrix} \leq \begin{bmatrix} -M \\ -M \\ -M \\ -M \\ -M \\ -M \end{bmatrix}$$

This ensures that $\mathbf{S}_i$ is greater than a for all nodes i.

Results and verification

The network is verified to have no paths from vulnerable to demand after removing edges.

The distance to detection in this design is 0 units.

Vulnerable Nodes	Time to sense	Sensor nodes	Actuator edges
1,2,3,19,32,37,39,53,66	0	19,39,53,66	1,13,27,30,38,39,40,44, 51,52,68,72,80,87,90,103,150,149
1,2,3	0	1,2,3	1,4,68
131,20	0	131	27, 30, 37

Table 2: The distance to detection for different sets of vulnerable nodes

In the previous work Palleti and Narasimhan (2016), the number of actuators suggested was 19 viz. 2, 12, 16, 24, 30, 38, 40, 48, 51, 52, 67, 71, 76, 86, 79, 89, 108, 110, 127 with 3 sensors (at nodes 40, 54, 67) and the same 9 vulnerable nodes.

Case 3: Vulnerable nodes may be attacked individually

Solution formulation

Formulating as mixed integer linear optimization problem:

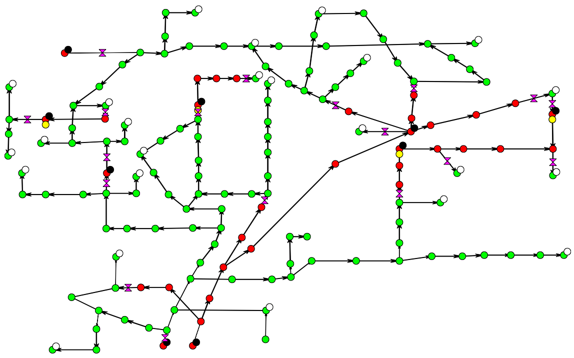

Figure 4: Case where all possible vulnerable nodes are always attacked simultaneously

Green – Demand partition

Red – Vulnerable partition

Black – Vulnerable nodes

White – Demand nodes

$$min \quad \left(\sum x_i + \sum z_i\right)$$

$$sub$$

$$\mathbf{A}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\\ \mathbf{d}\\ \mathbf{w}\\ \mathbf{v}\end{bmatrix} \leq \mathbf{b}$$

$$\mathbf{A_{eq}}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\\ \mathbf{z}\\ \mathbf{d}\\ \mathbf{w}\\ \mathbf{v}\end{bmatrix} = \mathbf{b_{eq}}$$

Where $\mathbf{x} \equiv [x_i]$ is a decision variable vector whose components are 1 if there exists a sensor at i^{th}node, 0 otherwise.

$\mathbf{y} \equiv [y_i]$ is a decision variable vector whose components are 1 if i^{th}node is in the demands side of the actuators(demand partition), 0 otherwise (source partition).

$\mathbf{z} \equiv [z_i]$ is a decision variable vector whose components are 1 if there exists an actuator at i^{th}edge, 0 otherwise.

$\mathbf{d} \equiv [d_i]$ is a decision variable vector whose components store $M-$the "distance to detection" for a particular vulnerable node.

$\mathbf{w} \equiv [w_i]$ is 1 if i^{th}node is in the demand partition, a large number (M) otherwise.

$\mathbf{v} \equiv [v_{ji}]$ is 1 if the j^{th} vulnerable node's distance to detection is inequal to the distance to the i^{th} (sensor2.1) node.

M is a large number greater in order than the distances in the network.

Hence the decision vector $\begin{bmatrix} \mathbf{x} & y & z & a & w & \mathbf{v} \end{bmatrix}$ is of length $3*N+E+V+N*V$, where V is the number of vulnerable nodes, N is the number of nodes, E is the number of edges.

$$A = \begin{bmatrix} A_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & A_2 & \mathbf{0} \\ & A_3 & \end{bmatrix}, A_{eq} = \begin{bmatrix} A_{eq1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & A_{eq2} & \mathbf{0} \\ & A_{eq3} & \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, b_{eq} = \begin{bmatrix} b_{eq1} \\ b_{eq2} \\ b_{eq3} \end{bmatrix}.$$

A_1 is a matrix of vulnerable nodes vs. affected nodes, with $A_{1ij} = -1$ if i^{th} vulnerable node affects the j^{th} node in graph.

$$b_1 = \begin{bmatrix} . \\ . \\ . \\ -1 \\ . \\ . \\ . \end{bmatrix}$$

$A_{eq1} = b_{eq1} = [\mathbf{O}]$

To link the partitioning variables $\mathbf{y}$ to the variables in the cost function, $\mathbf{z}$:

$A_2 = \left[\begin{array}{cc} J & I_{E\times E} \end{array} \right]$ where J is the incidence matrix.

$b_2 = [\mathbf{O}]$. The presence of an edge from source to demand forces the corresponding $\mathbf{z}_i$ to switch to one, indicating the presence of an actuator on that edge.

A_{eq2}, b_{eq2} provide bounds to force the decision variables to reality, viz. the demand nodes necessarily being in the demand partition and the vulnerable nodes being in the source. We might also include constraints to consider a few steps after vulnerable node to be necessarily in the source partition, as done in Palleti and Narasimhan (2016).

We use the shortest distances from each of the vulnerable nodes to all other nodes in the graph to model time to contamination. We're assuming the actual times are linear functions of distance. Let this matrix be $\mathbf{D}$. The first set of constraints of A_3 is one for each entry in $M - \mathbf{D}$ matrix, each of which is multiplied with $\mathbf{x}$. Each constraint is balanced by $-d_j$, where j is the vulnerable node under consideration in that constraint. $b_3 = [\mathbf{O}]$.

Now, d_j is greater than or equal to $M - D_{ij}$. We add constraints to force equality to the distance to the closest sensor: $D_{ij} + d_j - Mv_{ij} \leq M$ for all sensors. More constraints for making sure non-sensor nodes don't use positive v_{ij}: $-x_i + v_i \leq 0$. For ensuring at least one equality of D_{ij} and d_j, per j: $A_{1j}\mathbf{x} - A_j\mathbf{v}_j \leq -1$.

The actuators need to be at least $M - d_j$ distance away from the each of the vulnerable nodes. This can be implemented in two ways: constraining the actuators to be placed after a particular distance, or constraining the demand partition to start after a particular distance. Since generally speaking $N < E$ in a lot of networks, we show the node implementation.

Transform the $\mathbf{y}$ vector to $\mathbf{w}$, where $w_i = M$ denotes the sensor partition and $w_i = 1$ implies the node is in demand partition. Construct constraints for actuator placement for the i^{th}node as follows: $\begin{bmatrix} ... & 0 & -(M + D_{ij} - 1) & 0 & ... \end{bmatrix} [\mathbf{w}] + \begin{bmatrix} -1 \end{bmatrix} d_j \leq \begin{bmatrix} -2 * M \end{bmatrix}$. This ensures that when $\mathbf{w}_i$ is 1, the constraint is activated and only satisfied if the corresponding $\mathbf{D_{ij}} > \mathbf{d_j}$. When $\mathbf{w}_i$ is M, the constraint is always satisfied as $M^2$is several orders of magnitude greater than $2 * M$.

Small network example

Consider the same network in Figure 1, with the two vulnerable nodes (1, 2) and one demand node (6).

The additional constraints over case 2 introduced in this case are:

$$\begin{bmatrix} M-0 & 0 & 0 & 0 & 0 & 0 \\ 0 & M-1 & 0 & 0 & 0 & 0 \\ 0 & 0 & M-1 & 0 & 0 & 0 \\ 0 & 0 & 0 & M-2 & 0 & 0 \\ 0 & 0 & 0 & 0 & M-2 & 0 \\ 0 & 0 & 0 & 0 & 0 & M-3 \end{bmatrix} \mathbf{x} - \begin{bmatrix} d_1 \\ d_1 \\ d_1 \\ d_1 \\ d_1 \\ d_1 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} M-M & 0 & 0 & 0 & 0 & 0 \\ 0 & M-0 & 0 & 0 & 0 & 0 \\ 0 & 0 & M-1 & 0 & 0 & 0 \\ 0 & 0 & 0 & M-1 & 0 & 0 \\ 0 & 0 & 0 & 0 & M-1 & 0 \\ 0 & 0 & 0 & 0 & 0 & M-2 \end{bmatrix} \mathbf{x} - \begin{bmatrix} d_2 \\ d_2 \\ d_2 \\ d_2 \\ d_2 \\ d_2 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

This gets us variables storing $M - d_i$, the value after subtracting the minimum distance to sensor from the large number for each vulnerable node.

Constraints for ensuring the conditions around d_j's:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix} \mathbf{x} + \begin{bmatrix} d_1 \\ d_1 \\ d_1 \\ d_1 \\ d_1 \\ d_1 \end{bmatrix} - M \begin{bmatrix} v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \end{bmatrix} \leq M \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} M & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix} \mathbf{x} + \begin{bmatrix} d_2 \\ d_2 \\ d_2 \\ d_2 \\ d_2 \\ d_2 \end{bmatrix} - M \begin{bmatrix} v_{21} \\ v_{22} \\ v_{23} \\ v_{24} \\ v_{25} \\ v_{26} \end{bmatrix} \leq M \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

For v_i's are 1 only when i is a sensor:$-x_i + v_i \leq 0$. At least one equality constraint being satisfied:

$$\begin{bmatrix} -1 & -1 & -1 & -1 & -1 & -1 \\ 0 & -1 & -1 & -1 & -1 & -1 \end{bmatrix} \mathbf{x} - \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{v_1} \\ \mathbf{v_2} \end{bmatrix} \leq \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

The constraints for ensuring desired partitioning are:

$$
-1 \times \begin{bmatrix} M+0-1 & 0 & 0 & 0 & 0 & 0 \\ 0 & M+1-1 & 0 & 0 & 0 & 0 \\ 0 & 0 & M+1-1 & 0 & 0 & 0 \\ 0 & 0 & 0 & M+2-1 & 0 & 0 \\ 0 & 0 & 0 & 0 & M+2-1 & 0 \\ 0 & 0 & 0 & 0 & 0 & M+3-1 \end{bmatrix} \mathbf{w} - \begin{bmatrix} d_1 \\ d_1 \\ d_1 \\ d_1 \\ d_1 \\ d_1 \end{bmatrix} \leq
$$

$$
\begin{bmatrix} -2M \\ -2M \\ -2M \\ -2M \\ -2M \\ -2M \end{bmatrix}
$$

$$
-1 \times \begin{bmatrix} M+M-1 & 0 & 0 & 0 & 0 & 0 \\ 0 & M+0-1 & 0 & 0 & 0 & 0 \\ 0 & 0 & M+1-1 & 0 & 0 & 0 \\ 0 & 0 & 0 & M+1-1 & 0 & 0 \\ 0 & 0 & 0 & 0 & M+1-1 & 0 \\ 0 & 0 & 0 & 0 & 0 & M+2-1 \end{bmatrix} \mathbf{w} - \begin{bmatrix} d_2 \\ d_2 \\ d_2 \\ d_2 \\ d_2 \\ d_2 \end{bmatrix} \leq
$$

$$
\begin{bmatrix} -2M \\ -2M \\ -2M \\ -2M \\ -2M \\ -2M \end{bmatrix}
$$

Results and verification

The network is verified to have no paths from vulnerable to demand after removing edges. The distance to detection is always lesser than the distance to the first actuators.

Vulnerable Nodes	Time to sense	Sensor nodes	Actuator edges
1 2 3 19 32 37 39 53 66	0 0 0 0 0 0 0 0 0	1 2 3 19 32 37 39 53 66	1 4 40 48 51 52 68 150 151 152 153 149
1 2 3	0 0 0	1,2,3	1,4,68
1 2 3 19 32 37 39 53 66[1]	0 4 1 0 1 0 0 1 0	1 7 19 33 37 39 54 57 66	2 8 12 30 38 40 48 51 52 67 69 77 78 89 103 108 110

Table 3: The distance to detection for different sets of vulnerable nodes

In the previous work Palleti and Narasimhan (2016), the similar task was partial shutdown of WDN. The number of actuators suggested was 20 viz. 2, 12, 16, 24, 30, 38, 40, 48, 51, 52, 67, 71, 76, 86, 79, 89, 108, 110, 127, 112 with 3 sensors (at nodes 40, 54, 67) and the same 9 vulnerable nodes. The nodes directly downstream of vulnerable are taken as source nodes. Here for task 4, we get a solution with 26 cardinality: 17 actuators, 9 sensors. The previous work does not require that no demand node be compromised though.

3. Conclusions and future work

For a given WDN, we presented ways to arrive at a sensor and actuator placement strategy. This sensor and actuator network design is useful for water authorities to detect contaminants and isolate the contaminated area in the event of an attack on the WDN. We proposed a multi-objective linear programming formulation that gives the optimal sensor and actuator cardinality.

The formulation can reproduce the results of Palleti and Narasimhan (2016) after relaxing the full requirements to partial, viz. to prevent further contamination of demand nodes.

A performance benchmark of the simultaneous framework and the sequential non-linear optimization solution would be an illuminating exercise.

This formulation introduces an $O(NV)$ blowup in the number of decision variables, but there exist techniques to solve such specifically constrained variables much faster. An implementation using such techniques would be useful for assessing real performance.

Integration of an implementation of these algorithms with existing open-source toolboxes for sensor and actuator placement is a possible future action, which makes the techniques presented here much more accessible.

4. REFERENCES

Hart, W. E., Murray, R., 2010. Review of sensor placement strategies for contamination warning systems in drinking water distribution systems. Journal of Water Resources Planning and Management 136 (6), 611–619.
URL `http://dx.doi.org/10.1061/(ASCE)WR.1943-5452.0000081`

Kessler, A., Ostfeld, A., Sinai, G., 1998. Detecting accidental contaminations in municipal water networks. Journal of Water Resources Planning and Management 124 (4), 192–198.
URL `http://dx.doi.org/10.1061/(ASCE)0733-9496(1998)124:4(192)`

Ostfeld, A., Uber, J. G., Salomons, E., Berry, J. W., Hart, W. E., Phillips, C. A., Watson, J.-P., Dorini, G., Jonkergouw, P., Kapelan, Z., di Pierro, F., Khu, S.-T.,

Savic, D., Eliades, D., Polycarpou, M., Ghimire, S. R., Barkdoll, B. D., Gueli, R., Huang, J. J., McBean, E. A., James, W., Krause, A., Leskovec, J., Isovitsch, S., Xu, J., Guestrin, C., VanBriesen, J., Small, M., Fischbeck, P., Preis, A., Propato, M., Piller, O., Trachtman, G. B., Wu, Z. Y., Walski, T., 2008. The battle of the water sensor networks (bwsn): A design challenge for engineers and algorithms. Journal of Water Resources Planning and Management 134 (6), 556–568.
URL http://dx.doi.org/10.1061/(ASCE)0733-9496(2008)134:6(556)

Palleti, V. R., Narasimhan, S., 2016. Actuator network design to mitigate contamination in water distribution networks.

Palleti, V. R., Narasimhan, S., Rengaswamy, R., Teja, R., Bhallamudi, S. M., 2016. Sensor network design for contaminant detection and identification in water distribution networks. Computers & Chemical Engineering 87, 246 – 256.
URL http://www.sciencedirect.com/science/article/pii/S0098135415003865

Rathi, S., Gupta, R., 2014. Sensor placement methods for contamination detection in water distribution networks: A review. Procedia Engineering 89, 181 – 188.
URL http://www.sciencedirect.com/science/article/pii/S1877705814022905

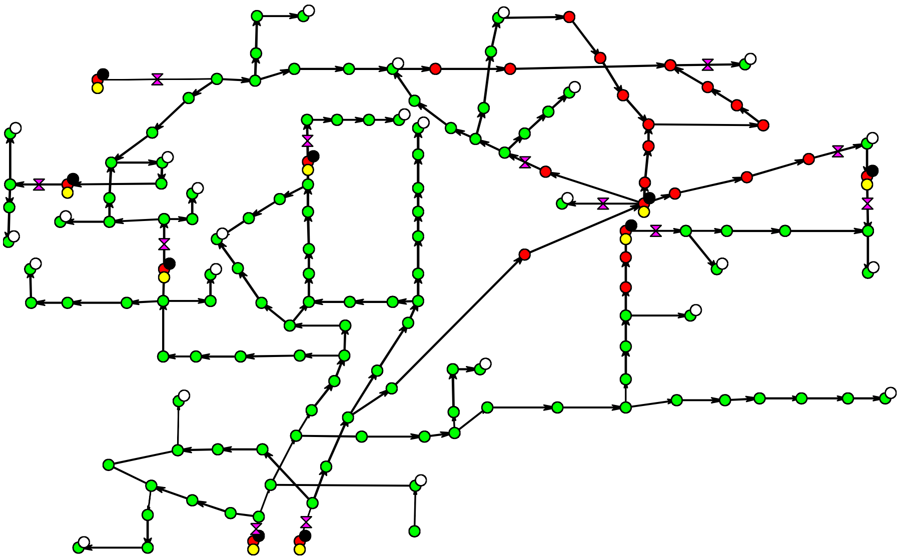

Figure 5: Sensor and actuator placement strategy for 9 vulnerable nodes

Green – Demand partition

Red – Vulnerable partition

Black – Vulnerable nodes

White – Demand nodes

Yellow – sensor nodes

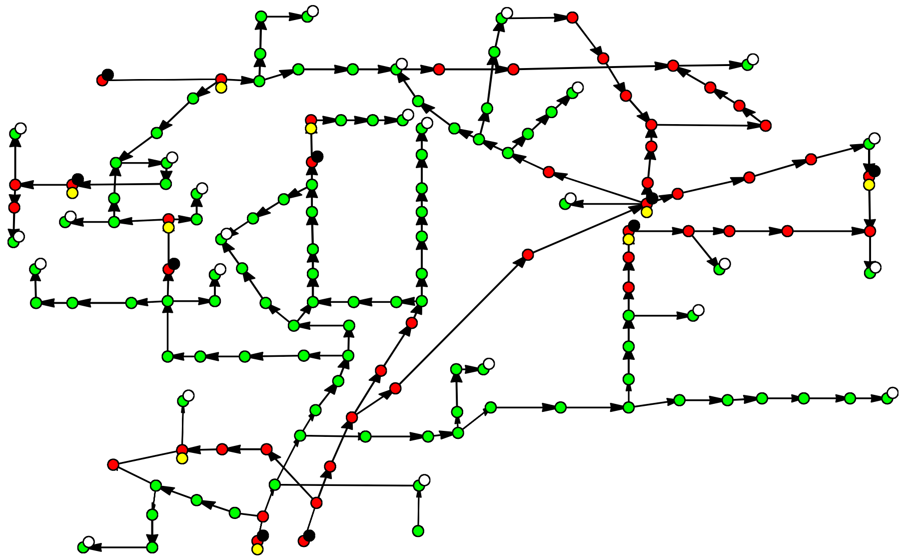

Figure 6: Sensor and actuator placement strategy for 9 vulnerable nodes and instant downstream contamination

Green – Demand partition

Red – Vulnerable partition

Black – Vulnerable nodes

White – Demand nodes

Yellow – sensor nodes

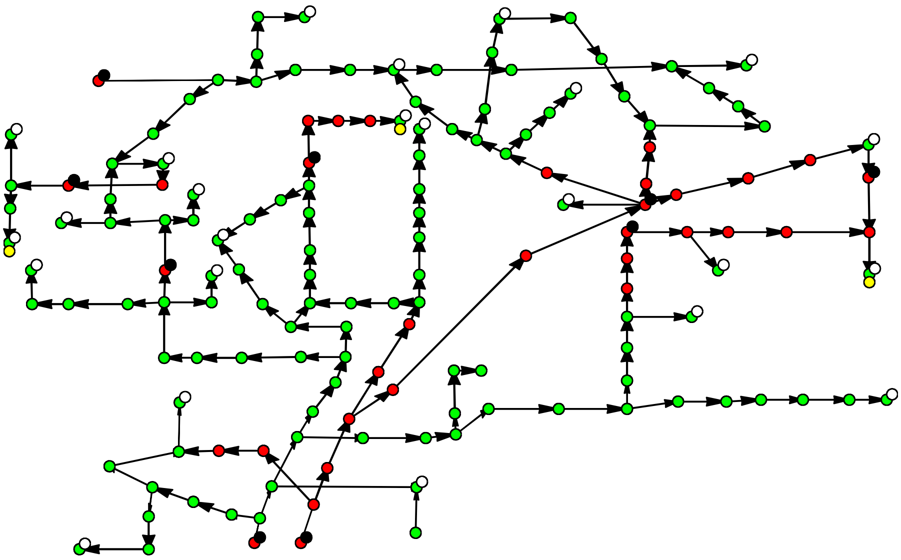

Figure 7: Validation of Palleti and Narasimhan (2016), case of allowing demand nodes to be compromised

The cardinality is 21, the same as presented in the earlier work.