

Abstract

Simultaneous sensor and actuator placement for identification and containment of contaminants in a water distribution network.

A multi objective integer linear optimization strategy assuming perfect sensors and actuators.

1 Scenario

1.1 Given

Specification of water distribution network – vulnerable nodes, demand nodes, the adjacency matrix.

Time-delay in sensors of contaminant sensing, lengths of pipes, accuracy of sensing, etc. can be added onto this work easily, and are ignored.

1.2 Requirements to be satisfied

To find distribution of sensors on nodes and actuators on edges such that the attack can be identified and the contaminant can be prevented from reaching the demands.

2 Previous work

Sensor placement using the principle that there must exist a unique non-zero set of sensors for each set of vulnerable nodes that can be affected. The work augmented the graph abstraction with vulnerable nodes representing multiple real nodes and used sets of affected nodes to force sensor placement. Actuator placement on edges to achieve a balanced min-cut, between the sensor nodes and demands, to contain contaminant[1]. It is also this work’s primary reference.

The approach in [3] attempts to optimize different objectives: maximizing detection likelihood and minimizing expected time to detection and solves it using genetic algorithms. This work assumes both of these objectives are binary – no time delay in detection and accurate sensing.

3 Hypotheses

Simultaneous sensor and actuator distribution can be achieved and is more efficient i.e. these are not independent problems, and these can be formulated as a ILP and solved.

4 Method

We first develop a formulation and algorithm for each case, implement in MATLAB, and compare with results from previous work to check veracity and im-

provement.

5 Implementation

Case 1: Shutting down the network effectively stops the contaminant beyond the actuator too.

In this simpler case, there are no additional constraints on the actuator placement problem beyond the being a min-cut of the entire graph. The sensor and actuator placement problems are independent.

As long as the sensor network can detect the contaminant before it reaches the demands and the actuation can happen simultaneously, the requirements are satisfied.

Formulation

A binary integer optimization problem is presented:

$$\begin{aligned} \min \quad & (\sum x_i + \sum z_i) \\ \text{sub} \quad & \\ \mathbf{A} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} & \leq \mathbf{b} \\ \mathbf{A}_{eq} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} & = \mathbf{b}_{eq} \end{aligned}$$

Where $\mathbf{x} \equiv [x_i]$ 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 $[\mathbf{x} \ \mathbf{y} \ \mathbf{z}]$ 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.

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

$$A_{eq1} = b_{eq1} = \mathbf{0}$$

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

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

A_{eq} , b_{eq} 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.

Verification

This was verified to produce the same results as [1].

Case 2: The contaminant contained only in the vulnerable side of actuator network

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

Formulation

Formulating as binary integer optimization problem:

$$\begin{array}{ll} \min & (\sum x_i + \sum z_i) \\ \text{sub} & \\ \mathbf{A} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} & \leq \mathbf{b} \\ \mathbf{A}_{eq} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} & = \mathbf{b}_{eq} \end{array}$$

Where $\mathbf{x} \equiv [x_i]$ 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.

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

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

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

$b_2 = [\mathbf{0}]$. This links the partitioning variables \mathbf{y} to the variables in the cost function, \mathbf{z} .

A_{eq} , b_{eq} 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.

After adding constraints from each of the sub-problems, the only thing left to do to enforce containment is to add constraints reflecting it. Our aim is to force whatever partitioning to happen in the region farther away from the contaminant than the sensor nodes. So 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”. There are still problems associated with this approach, which I’m working on now.

6 Example

For example, take the graph generated by $adjGraph = sparse([1 \ 1 \ 2 \ 2 \ 2 \ 3 \ 3 \ 4 \ 5], [2 \ 3 \ 4 \ 5 \ 3 \ 4 \ 5 \ 6 \ 6], [2 \ 2 \ 1 \ 1 \ 1 \ 1 \ 1 \ 2 \ 2], 6, 6);$

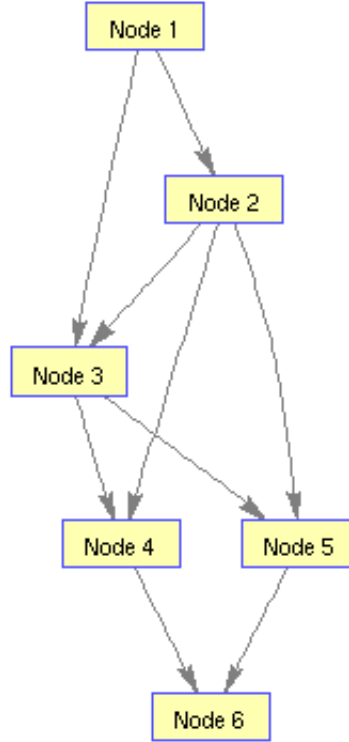


Figure 1: The test graph

I'm currently working on implementing this in MATLAB.

References

- [1] V. Reddy, 2015 - Sensor network design for contaminant detection and identification in water distribution networks.
- [2] On the structure of all min cuts in a network
- [3] Optimization of Contaminant Sensor Placement in Water Distribution Networks: Multi-Objective Approach
- [4] Review of Sensor Placement Strategies for Contamination Warning Systems in Drinking Water Distribution Systems 10.1061/ ASCE WR.1943-5452.0000081
- [5] Sensor Placement Methods for Contamination Detection in Water Distribution Networks: A Review