Optimizing Sensor Placement for Kentucky Pipe Network

Hayden Dessommes
Matthew Shaw
Aurimas Racas

ISYE 6333 - Operations Reseach for Supply Chain
Georgia Institute of Technology
Atlanta, Georgia

December 6, 2021

1 Problem background

In this project, we are asked to work on a problem posed by a water utility company, which is interested in allocating pressure sensors in order to detect pipe bursts in its water distribution network. The network is located in Kentucky, and is composed of 1123 pipes and 811 nodes. It provides 2.09 million gallons of water per day to its 5,010 customers.

The network operator possesses b pressure sensors that they wish to position on nodes of the network in order to detect pipe bursts. We are given information about the structure of the network which tells us which nodes, if equipped with a sensor, will be able to detect bursts at which pipes. The goal of this project is to determine the optimal location of the sensors to protect the network as best as possible.

2 Optimizing sensor location with equal pipe importance

2.1 Integer program formulation

First, we formulate an integer program that determines the location of b to be placed in such a way that maximizes the expected number of pipe bursts detected (where each pipe is assumed to have an independent burst probability of 0.1).

Our decision variables are:

- N_i - a binary variable that represents whether node i has a sensor installed (811 such variables are used, one for each node).
- P_j - a binary variable that represents whether pipe j will be detected by a sensor installed at one of the nodes (1123 such variables will be needed).

Indicator function $I(N_i, P_j)$ is created from the Detection_Matrix.csv file, which indicates the ability of node N_i (columns) to detect a burst in pipe P_j (rows). $I(N_i, P_j) = 1$ if and only if a sensor placed on node i can detect a burst in pipe j.

Our objective function is to maximize the number of expected bursts, where each pipe has a 0.1 burst probability:

$$Max \quad 0.1 \sum_{j=1}^{1123} P_j$$

The following constraints are required:

1. If a node i has a sensor installed, we need to make sure that P_j for all pipes j related to node i is equal to 1. This can be achieved with the following constraints for every pair of (i,j) where $I(i,j) = 1$:

$$P_j \geq N_i \quad \forall (i,j) | I(N_i, P_j) = 1$$

2. Similarly, we need to enforce that if none of the nodes that are related to pipe j have a sensor, then the burst in pipe j will be undetected. That can be expressed with the following constraints for every pipe j:

$$P_j \leq \sum_{i | I(N_i, P_j)=1} N_i \quad \forall j$$

3. We need to include a constraint that limits the number of sensors placed to a total number of sensors available b:

$$\sum_{i=1}^{811} N_i \leq b$$

4. As noted above, P_j and N_i are binary variables.

2.2 Results

We solved the optimization model described above using b values ranging from 1 to 20. Even with one sensor placed, we can detect 39 expected bursts. Only 19 sensors are required to cover the entire pipe network. We illustrate the detection capability as a function of number of sensors placed below.

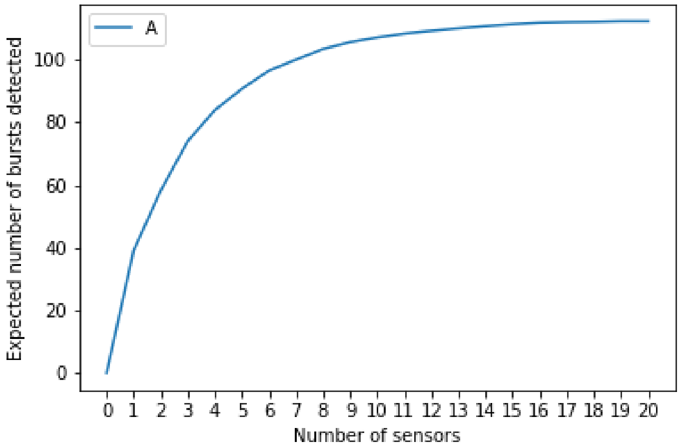

Figure 1: Burst detection capability as a function of sensors placed

2.3 Comparison with a greedy solution

We explore how the previously obtained solution compares to a greedy algorithm that simply picks the nodes that intersect with the most pipes for every sensor. That is, the following procedure is performed:

For 1..b where b is the number of sensors available:

- Find all the nodes N_i in the list of nodes N which do not yet have sensors placed on them (denoted as set N^*).
- Find all the pipes P_j in the list of nodes P which cannot be detected by previously placed sensors (denoted as set P^*).
- Among nodes in the set N^*, find the node N_i that maximizes the number of pipes in the set $P*$.

We illustrate the relative performance of the two approaches below. While the two approaches yield identical results for the first 3 sensors, the greedy placement strategy unperformed as the number of sensors increases, with the gap in the expected bursts to be detected approaching 10%.

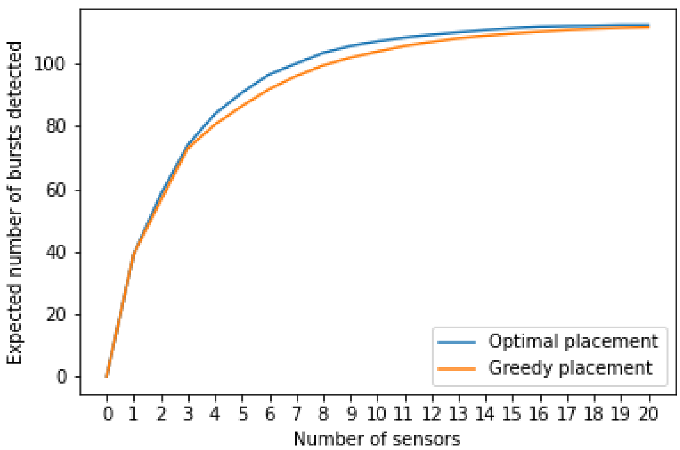

Figure 2: Burst detection capability as a function of sensors placed

3 Sensor placement - taking into account pipe criticality

In this section, we explore the optimal placement strategies if pipe criticality information is taken into account. Specifically, for every pipe P_j we are given a criticality index w_j in the range of $[0, 1]$. We are looking to develop an IP program that minimizes the highest criticality of a pipe that is not covered by a sensor.

3.1 Integer Program formulation

We are effectively solving a Min-Max problem with the following objective function and the same constraints as discussed in the previous section.

$$\min \max_{j \in [1;1123]} (1 - P_j) w_j$$

To linearize the above objective function, we introduce an additional decision variable z and thus our objective function becomes simply

$$\min z$$

With additional constraints for every j:

$$z \geq (1 - P_J) w_j \quad \forall j$$

Other constraints as described in section 2 are applicable, too, i.e.:

1. Pipe detection constraint:
$$P_j \geq N_i \quad \forall (i, j) | I(N_i, P_j) = 1$$

2. Pipe non-detection constraint:
$$P_j \leq \sum_{i | I(N_i, P_j) = 1} N_i \quad \forall j$$

3. Sensor limit constraint:
$$\sum_{i=1}^{811} N_i \leq b$$

3.2 Results

Below we present the results analysing the highest critical pipe left undetected by different number of sensors placed. There are 12 pipes with criticality of 1, and we find that it requires 4 sensors to be used before all such pipes are detected. It also appears that the network structure is such that it takes 14+ sensors before the maximum pipe criticality begins decreasing rapidly.

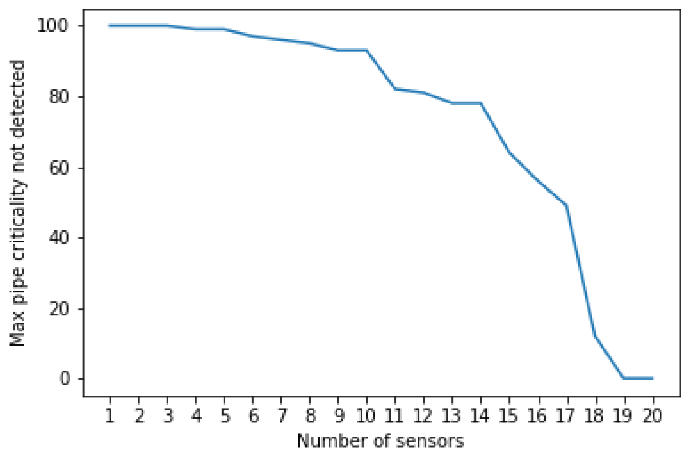

Figure 3: Max undetected criticality pipe as a function of number of sensors placed

3.3 Further improvements

While the above model and results satisfy the requirement stated (minimize the maximum undetected pipe criticality), we observe that the optimal values for less than 5 sensors are not what may be intended in the real world. In particular, we observed that the solver chooses a node that does not detect any pipes when only 1 sensor is placed, and it does not choose the most critical pipes until 4 sensors are introduced.

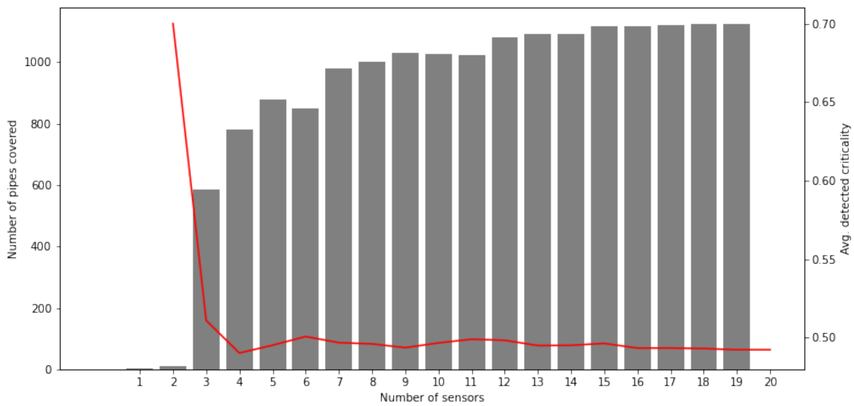

Figure 4: Detectable pipes as a function of number of sensors placed

From a mathematical point of view, this makes sense as the objective function value stays at 1 until all pipes with such criticality are covered, thus as long as not all of them are covered, it does not make a difference which pipes are chosen.

In real world, however, such behavior is not optimal. We thus introduce a slightly modified objective function that takes into account the average criticality of detectable pipes while ensuring that this does not impact the primary objective. The modified objective function is defined as follows:

$$\min z - \frac{1}{1123} \sum_{j=1}^{1123} P_j w_j$$

We use the total number of pipes 1123 as the denominator to ensure that the second component of the objective value stays in the range $[0, 1)$ and ensures that the overall objective value is predominantly decided by minimizing z. Note that we could not use a simple average criticality of detected pipes as this would make the objective function non-linear.

As illustrated below, the results obtained yield the overall objective function value identical to the previous one. However, as it can be seen in the second chart, the number of pipes covered by the sensors and the average criticality is different, especially for the cases where a small number of sensors are used.

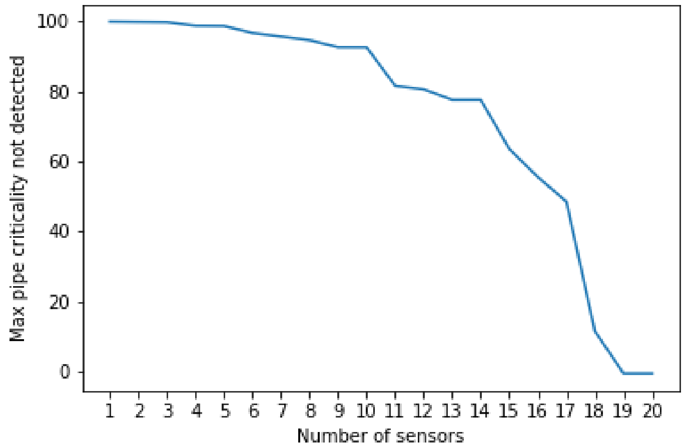

Figure 5: Max undetected criticality pipe as a function of number of sensors placed (modified objective)

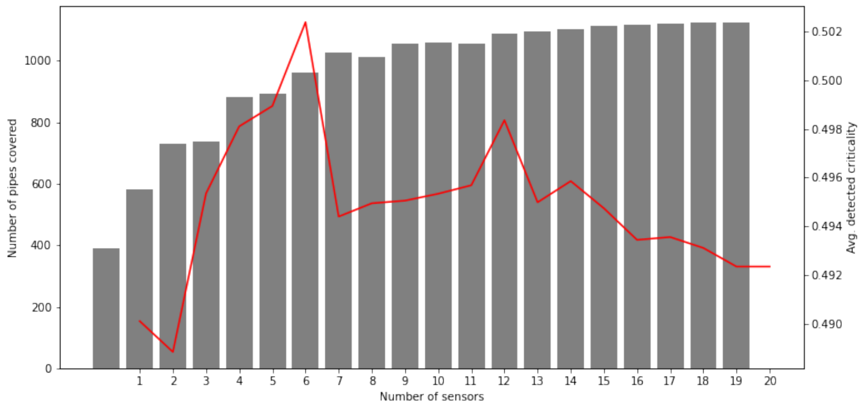

Figure 6: Detectable pipes as a function of number of sensors placed (modified objective)