Risk Aware Optimization of Water Sensor Placement

Antonio Candelieri †
Department of Economics, Management and Statistics
University of Milano-Bicocca
Milan, Italy
antonio.candelieri@unimib.it

Andrea Ponti
Department of Computer Science, Systems and Communication
University of Milano-Bicocca
Milan, Italy
a.ponti5@campus.unimib.it

Francesco Archetti
Department of Computer Science, Systems and Communication
University of Milano-Bicocca
Milan, Italy
francesco.archetti@unimib.it

ABSTRACT

This paper addresses the identification of optimal "sensing spots", within a network for monitoring the spread of "effects" triggered by "events". Many real-world problems fit into this general framework: we focused on the early detection of contamination events in Water Distribution Networks (WDN). We model the sensor placement as a bi-objective optimization problem, aiming at minimizing the mean and standard deviation of detection time over a set of different simulated contamination events and solved using NSGA-II. A problem-specific data structure is proposed enabling a deeper analysis of empirical convergence of the population.

CCS CONCEPTS

• **Computing methodologies** → Artificial Intelligence, Search methodologies; • **Applied computing** → Operations research

KEYWORDS

Sensor placement, water network, multi-objective optimization, evolutionary optimization

ACM Reference format:

Antonio Candelieri. Andrea Ponti and Francesco Archetti. 2021. Risk aware Optimization of Water Sensor Placement. In *Proceedings of the Genetic and Evolutionary Computation Conference 2021 (GECCO '21). ACM, New York, NY, USA, 2 pages* https://doi.org/10.1145/3449726.3459477

1 Introduction

A WDN is modelled as a graph $G = (V, P)$ where vertices in V are junctions, tanks, reservoirs or consumption points, and edges in P are pipes, pumps, and valves. We assume a set of possible locations for placing sensors, that is $L \subseteq V$. Thus, a Sensor Placement (SP) is a subset of sensor locations, with the subset's size less or equal to p depending on the available budget. An SP is represented by a binary vector $s = \{0,1\}^{|L|}$ with $s_i = 1$ if a sensor is placed at node i, $s_i = 0$ otherwise, and $\sum s_i = p$. Let A denote the set of contamination events. Then, a probability distribution is placed over possible contamination events. In the computations we assume – as usual in the literature – a uniform distribution, but other discrete distributions are also possible. The objective $f_1(s)$ is computed as the average of detection time $f_1(s) = \frac{1}{|A|}\sum_{a\in A}\hat{t}_a(s)$, with $\hat{t}_a$ the minimum time step at which concentration, at one of the sensors in s, reaches or exceeds a given threshold τ for the event $a \in A \subseteq V$. The second objective is the standard deviation of the detection time $f_2(s) = \sqrt{\frac{1}{|A|}\sum_{a\in A}\left(\hat{t}_a(s) - f_1(s)\right)^2}$. Both $f_1(s)$ and $f_2(s)$ are minimized and for computing them we have used the Python package Water Network Tool for Resilience (WNTR) [1].

2 Single Sensor and Sensor Placement Matrices

Let S^ℓ denotes the so-called *sensor matrix*, with $\ell = 1, \dots, |L|$ an index for the location where the sensor is deployed at. Each entry s^ℓ_{ta} is the concentration of the contaminant for the event $a \in A$ at the simulation step $t = 0, .., K$, with $T_{\max} = K\Delta t$. In our study $T_{max} = 24$, $\Delta t = 1$ and $K = 24$. Without loss of generality, we assume that the contaminant is injected at the begin of the simulation. A *sensor placement matrix*, $H^{(s)} \in \mathbb{R}^{(K+1)\times|A|}$ is also defined, where every entry h_{ta} is the maximum concentration over those detected by the sensors in s, for the event a and at time step t. Suppose to have a SP s consisting of p sensors with associated *sensor matrices* $S^1, \dots, S^p$, then $h_{ta} = \max_{j=1,\dots,p} s^j_{ta}$ $\forall a \in A$. Figure 1 shows the SP matrix of a SP consisting of 2 sensors.

There is a relation between s and $H^{(s)}$: the columns of $H^{(s)}$ having maximum concentration at row $t = 0$ (i.e., injection time) are those associated to events with injection occurring at the deployment locations of the sensors in s. $H^{(s)}$ is the basic data structure on which detection times are computed: $\hat{t}_a$ is the minimum time step at which concentration reaches or exceeds a given threshold τ for the scenario a, that is $\hat{t}_a = \min_{t=1,\dots,K}\{h_{ta} \geq \tau\}$.

† Corresponding author

GECCO '21 Companion, July 10–14, 2021, Lille, France
 ACM ISBN 978-1-4503-8351-6/21/07. https://doi.org/10.1145/3449726.3459477

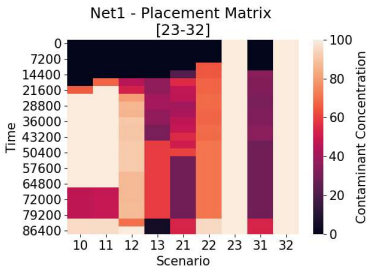

Figure 1: An example of sensor placement matrix

3 Search space and information space

The search space consists of all the possible SPs, that is $s \in \Omega \subseteq \{0,1\}^{|L|}$, satisfying the constraint $\sum s_i = p$. The computation of the two objectives $f_1(s)$ and $f_2(s)$ requires the generation of the associated sensor placement matrix $H^{(s)}$. For the sake of simplicity, let us denote with π this computational process:

$$s \xrightarrow{\pi} H^{(s)} \implies \phi(H^{(s)}) = (f_1(s), f_2(s))$$

We use $\phi(H^{(s)})$ to stress the fact that the computation is performed over $H^{(s)}$ – within the "information space" – and then it generates the observation of the two objectives $(f_1(s), f_2(s))$. A graphical representation of the mapping is given in Figure 2.

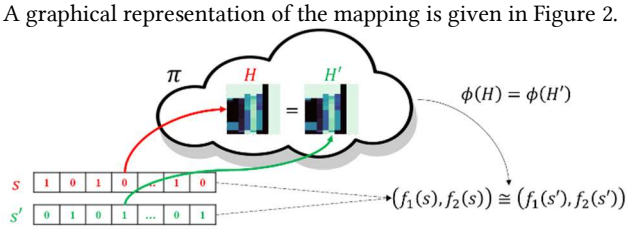

Figure 2: Search space, information space and objectives

Two SPs that are distant in Ω might correspond to a similar sensor placement matrix, leading to very close objectives values. Suppose to have $s, s' : d(x, x') = d_{max}$ then you could anyway observe $(f_1(s), f_2(s)) \cong (f_1(s'), f_2(s'))$ if $\delta(H, H') \cong 0$, with $H = H^{(s)}$, $H' = H^{(s')}$ and $\delta(.,.)$ a suitable distance between matrices (in this paper $d(x, x')$ is the Hamming distance and $\delta(H, H')$ is the Frobenius Norm of $H - H'$). This means that the landscape of the problem may have a huge number of global (not only local) optima, also significantly distant among them in Ω. Thus, the convergence in terms of objectives does not translate into a convergence in terms of the homogeneity of the population. We considered the Kappa criterion [2], based on the quotient of the sum of all normalized distances between all individuals: $k_{max} = (\mu^2 - \mu)/2$, with μ the population size. Since the optimization framework used in this paper – i.e., Pymoo (Python multi-objective optimization) [3] and specifically the NSGA-II algorithm [4] – allows to customize both operators and termination criteria, we extended the performance indicators with two Kappa measures: one computed in the search space (Kappa-Hamming) and one in the information space (Kappa-Frobenius).

4 Computational results

Neptun is a small WDN in Timisoara, Romania [5]. Its associated graph consists of 333 nodes and 339 edges (not reported here for space limitations). We set $|L| = |A| = 332$ (i.e., we have excluded the reservoir) and assumed $p = 25$ as the maximum number of sensors in a SP. We have analysed: (*i*) Hypervolume, (*ii*) Number of individuals belonging to the approximate Pareto Set, (*iii*) Kappa-Hamming and (*iv*) Kappa-Frobenius. The NSGA-II settings: population size equal to 40, 500 generations and 30 independent runs (all the other NSGA-II's setting were set to the default values). Figure 3 (left) shows both the hypervolume of the approximated Pareto frontier and the number of individuals in the population belonging to the approximated Pareto set, at each generation. Values are given as mean and standard deviation on 30 independent runs. Analogously, Figure 3 (right) reports the Kappa-Hamming and Kappa-Frobenius indicators over generations. While the population becomes more homogeneous within the search space (i.e., Kappa-Hamming decreases along generations), this is not true in the information space. Moreover, Kappa-Frobenius is larger than Kappa-Hamming – we want to remark that Kappa is a normalized indicator ranging in [0,1], therefore the two indicators can be compared, even if the base distances vary in two different ranges.

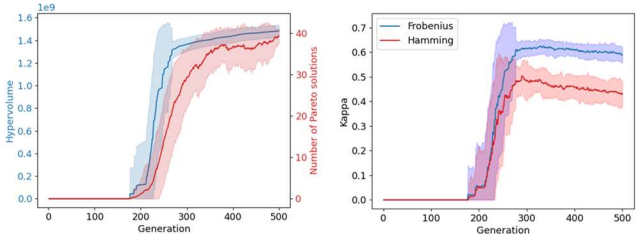

Figure 3: Left: hypervolume (blue) and number of Pareto individuals (red). Right: Kappa indicators at each generation

As main result, the comparison between the two Kappa indicators is a reliable performance metric of the optimization process and allows to also detect a lack of convergence in the information space. The proposed approach is applicable to other problems sharing a network structure with spatiotemporal dynamics data, such as fake news detection is social networks [6].

REFERENCES

[1] Klise, K. A., Murray, R., & Haxton, T. (2018). An Overview of the Water Network Tool for Resilience (WNTR) (No. SAND2018-4462C). Sandia National Lab. (SNL-NM), Albuquerque, NM (United States).

[2] Ghoreishi, S. N., Clausen, A., & Joergensen, B. N. (2017). Termination Criteria in Evolutionary Algorithms: A Survey. In IJCCI (pp. 373-384)

[3] Blank, J., & Deb, K. (2020). pymoo: Multi-objective Optimization in Python. IEEE Access, 8, 89497-89509.

[4] Deb, K. (2002). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-2. IEEE Trans. Evol. Comput., 6(2), 182-197.

[5] Candelieri, A., Soldi, D., & Archetti, F. (2015). Cost-effective sensors placement and leak localization–the Neptun pilot of the ICeWater project. Journal of Water Supply: Research and Technology—AQUA, 64(5), 567-582.

[6] Amoruso, M., Anello, D., Auletta, V., Cerulli, R., Ferraioli, D., & Raiconi, A. (2020). Contrasting the spread of misinformation in online social networks. Journal of Artificial Intelligence Research, 69, 847-879.Conference Short Name: WOODSTOCK'18.