Artificial Intelligence (AI) Contribution to GIS in Optimal Positioning of Hydrophone Sensors Using Genetic Algorithm (Case Study: Water Network, Casablanca, Morocco)

Hatim Lechgar, Abdelouahed Mallouk, Mohamed El Imame Malaainine, Tarik Nahhal and Hassane Rhinane

Abstract All over the world, potable water networks suffer from substantial leaking which may occur in places that are hard to locate. In addition to economic loss, there is potential hazard of epidemics. Thus, real time detection and hence fixing water leaks is essential for an efficient distribution network. Hydrophone sensors come in handy thanks to their ability to detect leaks. Nevertheless, a comprehensive covering of the network is impossible financially. This leads to searching for an arrangement of sensors to have an optimal coverage of distribution network. Linear search for optimal solutions in big networks breaks into an explosion of calculations which is unpractical in terms of resources allocation and time execution. We present how artificial intelligence (AI), and especially genetic algorithms (GA), offer both efficient and fast ways of figuring out the best disposition of sensors for an optimal coverage of the water network.

Keywords GIS · Artificial intelligence · Network's sensors · Network coverage · Genetic algorithm · SLOTS

H. Lechgar (✉) · A. Mallouk · M.E.I. Malaainine · T. Nahhal · H. Rhinane
Faculté Des Sciences Ain Chock, Université Hassan II-Casablanca, Casablanca, Morocco
e-mail: h.lechgar@gmail.com

A. Mallouk
e-mail: wahid.mallouk@gmail.com

M.E.I. Malaainine
e-mail: mohamed.malaainine@gmail.com

T. Nahhal
e-mail: t.nahhal@fsac.ac.ma

H. Rhinane
e-mail: h.rhinane@gmail.com

A. El Oualkadi et al. (eds.), *Proceedings of the Mediterranean Conference on Information & Communication Technologies 2015*, Lecture Notes in Electrical Engineering 381, DOI 10.1007/978-3-319-30298-0_2

1 Introduction

GIS problems are often subject to what is known as curse of dimensionality, which means that the state space grows rapidly when the number of parameters increases. However, the use of intelligent algorithms reduces considerably the size of the state space and helps to quickly find the optimal configurations.

In this chapter we will study the case of optimization of hydrophone sensors placement in a water network. Hydrophone sensors are devices that could listen to noise along a certain distance in water pipe.

There are several approaches of optimizing placement that have been used, especially in the problem of the Battle of the Water Sensor Networks (BWSN) [1], which studies the optimization of placement of sensors that detects a contamination in a water network. Other studies related to flaws detection apply other methods using pressure drop sensors [2]. Nevertheless pressure is not quite trustworthy unless to detect minor bursts, due to daily fluctuation and variation which enables spotting only major flaws.

We will show in this chapter the application of the methods used in BWSN, in the positioning of Hydrophone sensors namely greedy, and slots algorithms, and use their results as inputs to an elaborated Genetic algorithm.

1.1 Problem Formulation

Hydrophone sensors are devices placed on valves of water network and can listen to noise along a specific distance in a pipe, and hence allow the detection of an abnormal high noise; which is correlated with "bursts or leaks in pipes" [3].

An exhaustive coverage of the network is financially impossible, so water distribution companies tend to place sensors only on areas of recurrent burst. However, using sequential search techniques to maximize the coverage of a network or at least achieve a high percentage, bursts rapidly into what is known as combinatorial explosion. For instance the computing of all possible configurations (state space) of 20 sensors in a small subset of water network that contains 130 valves (possible placements) would take 3000 years even with a computer that does 1 trillion operations per second.

$$\binom{\mathbf{130}}{\mathbf{20}} / (\mathbf{10}^{12} * \mathbf{3600} * \mathbf{24} * \mathbf{365}) \cong \mathbf{3000\,years}) \tag{1}$$

It is worth noting also that even with a complete coverage of valves there may be still some dark areas of network that could not possibly be listened to. So we had to restrict our study to the subset of water network that is listenable.

We formulated the problem as follows:

- Let **NetW** be a network of water with **n** valves.
- Let **m** be the number of sensors.
- Let **TargPrc** be the target percentage of covered pipes.
- Find the minimum number of sensor **m** and their optimal placement on **n** candidate valves in order to reach at least **TargPrc** in **NetW**.

By solving this problem we guarantee that we can detect leaks across the wanted percentage of water network with **m** sensors.

1.2 Related Work

There are several works in similar problems, especially regarding BWNS [4]. Other problems based on calculating pressure drop to detect leaks, use structural analysis and integer optimization [2]. However acoustic based detection is more reliable thanks to its better sensitivity to abnormal noise. Casillas et al. (2013) continue in the same optic of using integer optimization and enhanced the performance with genetic algorithm [5]. Nevertheless there are other techniques that may be less optimal but more straightforward and efficient. For instance SLOTS is the best non evolutionary algorithm that solves the BWNS [6]. Genetic algorithm comes in handy because it can optimize far more initial suboptimal solutions. So we can use outputs from a greedy algorithm or SLOTS and optimize them afterward in an evolutionary way.

1.3 Overview of Our Method

We proposed to apply at first greedy algorithm and SLOTS algorithm to compare their results and efficiency then opt for the two best results to incorporate in a genetic algorithm.

2 Data Preparation

2.1 Subset of Listenable Network

We restricted the study domain to the reachable pipes. The reachable pipes are pipes that could be listened to while having a sensor on each valve in the network.

2.2 Cutting to Elementary Segments

Processing segments in their variety of length could slow down computation because that would imply for example cutting a pipe to the distance it is heard by a sensor, and hence write in the spatial database 2 new segments. To avoid this trap, we split all segments into equi-length segments of size **d**, so we did not have to bother about the geometry or length of each segment at once. This resulted in a bigger segment table, but the speed tradeoff was worthwhile.

2.3 Equivalent Homogenous Network

A hydrophone sensor come with a manufacturer note on how long it can listen across a specific materiel. For example the sensor could listen up to a distance '**a**' along a steel pipe, while it could listen to a different distance '**b**' along concrete one. That adds a complexity to our problem, because computations would require checking the type of each pipe to figure out how much length it could be listened to.

To reduce that complexity, we transformed the real length $\mathbf{LP_{real}}$ of each pipe to an equivalent length $\mathbf{LP_{eq}}$ as follows:

- Let **SeP** be the sensor sensitivity to the materiel of pipe **P**.
- Let $\mathbf{Se_{min}}$ be the lowest sensor sensitivity to all materiels.
- And $\mathbf{KP = SeP/Se_{min}}$

So:

$$\mathbf{LP_{eq}} = \mathbf{KP} * \mathbf{LP_{real}}. \tag{2}$$

2.4 Adjacency Tables

Spatial data are inherently well formatted and indexed in spatial databases.

However a massive querying in problems like the one we are tackling could make the computation process inefficient.

That is why we opted for the use of the representation of our network as an attribute table rather than attacking spatial entities directly. A common representation of this table is adjacency table, in which we correspond to each valve, all correspondent network (elementary) segments and to each segment its adjacent segments. These generated tables were then indexed, and we could acknowledge the evident enhancement in queries speed.

2.5 Preparation of Initial Heuristic Table

A heuristic is a function that estimates at each state, in a search problem space, the proximity to the goal state.

In order to implement AI search techniques in our problem, we built a table of heuristics in which each valve (a possible sensor position) is matched to the sum of all lengths of the pipes that can detect.

Each time a segment is matched to a sensor (that is a valve position) it gets subtracted from all other positions that could reach it.

This heuristic is used while searching for the next state in graph search both in greedy and SLOTS.

3 Methodological Approach

The couple {**m**,**TargPrc**} presents a trade-off. That is we don't know what the maximum achievable **TargPrc** with **m** sensors is. So in order not to complexify the problem, we applied for each number **m** of sensors both Greedy algorithm and SLOTS to compare their results. Afterwards we chose two outputs of better-off method to plug in Genetic algorithm as inputs.

At the end, it is the job of the decision-maker to determine which couple is affordable.

3.1 Greedy Algorithm

We applied at first the greedy algorithm. The greedy algorithm uses mainly the heuristic we built. It is simple but can be very suboptimal because it does not question at any step its previous choices.

Greedy algorithm is as follows:

- Let **N** be the number of sensors to be placed and **Sj** the sensor at location **(j)**.
- The greedy Algorithm proceeds in **N** iterations by choosing each time a **Sj** that enables the sensor to listen to the maximum length of network (our heuristic), and adds its location to the set of solution.

3.2 SLOTS Algorithm

SLOTS stands for Sensors local optimal transformation system by Dorini et al. (2010) [7]. It was developed in response to the BWNS and it is as follows:

- Begins with an arbitrary set of **m** sensor locations.
- One sensor moves while other sensors are held fixed, performance changes are measured each time.
- The sensor is placed in the location corresponding to the best improvement in performance.
- The algorithm terminates when there is no improvement from loop to loop.

3.3 Genetic Algorithm

Genetic algorithm (GA) is a search technique that mimics the process of natural selection [8].

To prevent the algorithm from converging on suboptimal local minima or maxima, an initial population of 'parent' solutions is first generated (Huang et al. 2006) [9].

That is why to process GA for **m** sensors we applied SLOTS to **m** sensor with 2 arbitrary initial locations, and use their respective results as the initial individual in GA.

- Let **Confg** be a set of bits such as its cardinal is equal to the number of valves in the network. **Confg** stands for configuration.
- Let **bi** be the bit that corresponds to the valve number **i** and **nbSns** the number of sensors.

So:

$$\mathbf{Confg} = \{\mathbf{b1}, \mathbf{b2}, \ldots, \mathbf{bn}\}. \tag{3}$$

$$\mathbf{bi} = \begin{cases} \mathbf{0} & \textbf{if no sensor is placed on it}. \\ \mathbf{1} & \textbf{if a sensor is placed on it}. \end{cases} \tag{4}$$

$$\sum_{\mathbf{1}}^{n} bi = \mathbf{nbSns} \tag{5}$$

Each state in the search space of our problem is consequently represented by a different configuration.

The Genetic algorithm goes as follows:

- 2 initial random propositions, said individuals, are picked.
- The configuration, or in GA terminology chromosome, of each individual is **Confg**.
- A utility function called 'fitness function' calculates for each individual its total coverage.

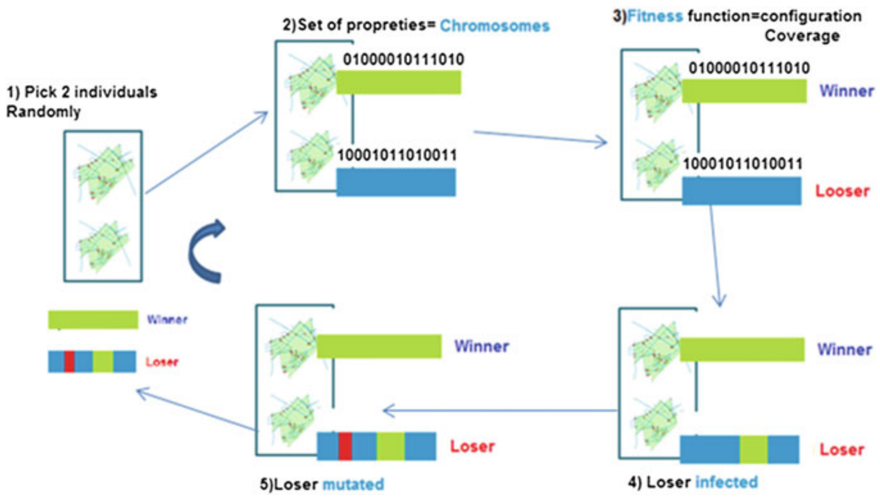

Fig. 1 GA is a search technique that mimics the competition and mutations in evolution theory. The process goes on again and again until hitting the maximum allocated time

- The configuration that gets the best coverage is the winner while the other one is declared looser.
- The looser configuration gets infected randomly by 10 % of the winner chromosomes.
- The looser get 5 % of its chromosome changed or muted.
- And the computations go on again until hitting the maximum allocated time (Fig. 1).

4 Case Study: Sensing a Water Network in Casablanca

4.1 Study Area and Settings

We chose for test, a medium sized subset of Casablanca water network with 130 valves and 13 km of pipes. We run firstly the greedy algorithm then 2 SLOTS with different initial configurations. The outputs of SLOTS were plugged in as inputs in GA (Fig. 2).

4.2 Results

As expected, for each number **m** of sensors, the SLOTS algorithm outperformed the greedy algorithm while the GA outran both of them.

Nevertheless, to achieve high coverage percentage we had to allocate a fair amount of time (2 h) for every **m** sensors case (Fig. 3).

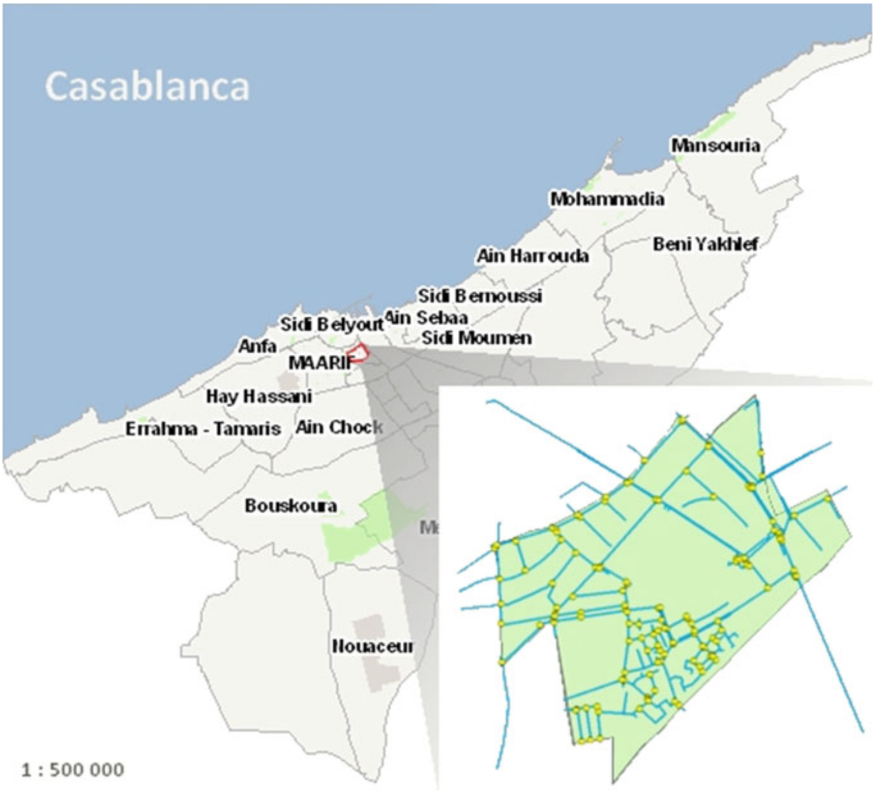

Fig. 2 A subset of Casablanca water network, extracted from GIS database, with 130 valves and 13 km of pipes

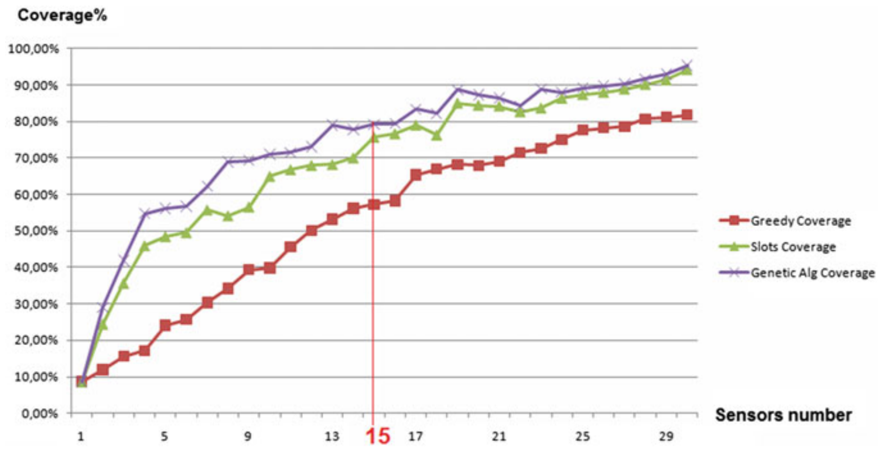

Fig. 3 The SLOTS curve represents the best results of the 2 SLOTS results. With 15 sensors we achieved 57 % coverage with greedy, 76 % with SLOTS and 80 % with GA

5 Conclusion and Further Thoughts

AI search techniques enable a fast and efficient way to optimize placement of spatial entities in a network. Furthermore, Genetic algorithm offers to optimize already optimized solutions.

While multithreading would be a good alternative to process simultaneously computations for different numbers of sensors, more sophisticated heuristics could however accelerate both SLOTS and GA.

References

1. Ostfeld, A., et al.: The battle of the water sensor networks (BWSN): a design challenge for engineers and algorithms. J. Water Resour. Plann. Manag. **134**(6), 556–568 (2008)
2. Sarrate, R., Nejjari, F., Rosich, A.: Sensor placement for fault diagnosis performance maximization in distribution networks. In: 20th Mediterranean Conference on Control and Automation (MED), 2012. IEEE (2012)
3. Hunaidi, O., et al.: Detecting leaks. J. AWWA **92(**2), 82–94 (2000)
4. Thompson, K.E., et al.: Optimal macro-location methods for sensor placement in urban water systems. Technology **37(**1), 205–213 (2005)
5. Casillas, M.V., et al.: Optimal sensor placement for leak location in water distribution networks using genetic algorithms. Sensors **13(**11), 14984–15005 (2013)
6. Thompson, K.E., et al.: Optimal macro-location methods for sensor placement in urban water systems. Technology **37(**1), 72–75 (2005)
7. Dorini, G., et al.: SLOTS: effective algorithm for sensor placement in water distribution systems. J. Water Resour. Plann. Manag. **136**(6), 620–628 (2010)
8. Mitchell, R.J., Chambers, B., Anderson, A.P.: Array pattern control in the complex plane optimised by a genetic algorithm. In: Tenth International Conference on Antennas and Propagation (Conf. Publ. No. 436). vol. 1. IET (1997)
9. Huang, J.J., McBean, E.A., James, W.: Multi-objective optimization for monitoring sensor placement in water distribution systems. In: 8th Annual Symposium on Water Distribution Systems Analysis. Environmental and Water Resources Institute of ASCE (EWRI of ASCE), New York (2006)

http://www.springer.com/978-3-319-30296-6

Proceedings of the Mediterranean Conference on Information & Communication Technologies 2015
MedCT 2015 Volume 2
El Oualkadi, A.; Choubani, F.; El Moussati, A. (Eds.)
2016, XIV, 692 p. 270 illus., 176 illus. in color., Hardcover
ISBN: 978-3-319-30296-6