

# A new methodology for the optimal design of series of pipes in sewer systems

Natalia Duque, Daniel Duque and Juan Saldarriaga

## ABSTRACT

The sewer network design problem consists of determining both the layout and the hydraulic design of the system. This paper aims to find an optimal hydraulic design for a specific layout consisting of a series of pipes. An optimal hydraulic design of a series of pipes is that which satisfies all the hydraulic, commercial, and construction constraints, while minimizing the construction costs. The present paper proposes a graph modeling framework in which the result of a shortest path problem coincides with the hydraulic design, and the underlying graph models the diameter and slope of each pipe in the series. To assess the performance of the methodology, several numerical examples are presented varying the pipe material, the topography, and the number of pipes in the series.

**Key words** | design of series of pipes, graph modeling, optimization, sewer systems, shortest path problem

### Natalia Duque

Juan Saldarriaga (corresponding author)  
Centro de Investigaciones en Acueductos y  
Alcantarillados (CIACUA), Departamento de  
Ingeniería Civil y Ambiental,  
Universidad de los Andes, Bogotá,  
Colombia  
E-mail: jsaldarr@uniandes.edu.co

### Daniel Duque

Centro para la Optimización y Probabilidad  
Aplicada (COPA), Departamento de Ingeniería  
Industrial,  
Universidad de los Andes, Bogotá,  
Colombia

## INTRODUCTION

The sewer network design problem consists of determining both the layout and the hydraulic design of the system. The layout of the network is defined by the initial pipes and the flow direction of each pipe. The hydraulic design determines the diameter and the slope of the pipe to be installed at each link of the sewer network, where a link refers to the span between two manholes. Moreover, pipe diameters are chosen from a discrete set of commercial diameters and the slope of a pipe is related to the elevation gap of its extremes. In this paper, the layout is assumed to be a series of pipes that might belong to a complete sewer network. The challenge is to find a minimum-cost design that accomplishes all hydraulic and construction constraints established by the corresponding national legislation. In the following, it is assumed that the design flow rate for each pipe is known beforehand. The design flow rate for a pipe corresponds to the inflow at the upstream manhole of the pipe plus the flow rate coming from the upstream pipes. The input information of the problem includes topographic information (horizontal length of the links and ground elevation of the manholes), commercial characteristics (available pipe materials and diameters),

physical characteristics of the fluid (water density and viscosity), and hydraulic characteristics (flow resistance formula and pipe's internal roughness). In this work, the hydraulic constraints that must be fulfilled are: a minimum pipe diameter, a maximum filling ratio, a minimum wall shear stress, a minimum and maximum velocity, and a minimum and maximum slope.

Sewer network design involves two different problems (the layout definition and the hydraulic design), and due to their complexity it is not common practice to solve them simultaneously. Available solution strategies in the literature propose solving both problems sequentially, but the hydraulic design problem has received more attention through both exact and heuristic methodologies.

Several exact approaches have been proposed for both problems. Haith (1966) used dynamic programming (DP) to obtain an optimal design of a series of pipes by considering a single pipeline divided into different segments, each with constant inflow and cost parameters. Decision variables in the model include the invert elevation at the end of each pipe and its diameter. However, this methodology had computational limitations due to the technology at that time.

Kulkarni & Khanna (1985) also used a DP approach to design a minimum-cost gravity-pumped sewer network using a modification of the Hazen–Williams equation. The authors simplified the design problem based on the concept of cost-effective feasible groupings at junctions. This DP approach consists of two parts: the first one fixes control variables associated with each link, and the second one calculates the hydraulic and cost statements. DP methodologies often suffer from the well-known curse of dimensionality, which limits their ability to solve large-scale instances. Li & Matthew (1990) used a nonlinear programming model to establish both topographic and hydraulic factors such as flow rate, size, and gradient of pipes. In this methodology, the authors divided the problem into two submodules. Given a layout, the first submodule optimizes the size and gradient of the pipes and the location of on-line pumping stations. The second submodule determines the flow rate using a searching direction method that looks for pipes where the flow rate can be modified to decrease the value of the objective function. To set the initial layout, Dijkstra's algorithm is used to create a shortest path-spanning tree, with the links weight being the product between its length and its average ground elevation. More recently, Swamee & Sharma (2013) used linear programming (LP) to obtain an optimal design for a series of pipes without linearizing the objective function or the constraints. To do so, lengths and diameters of each pipe are fixed to the commercially available diameters. The authors tested their methodology over a series of five pipes and obtained an optimal (yet simplified) design.

Heuristic approaches have also been proposed for this design problem. Holland (1966) used a general optimization routine to design a sewer network. This optimization routine performs a random search that works with continuous diameters, which are then rounded to the closest commercial diameter available. Elimam *et al.* (1989) used an LP-based heuristic approach, which relies on a piecewise linearization of the nonlinear terms such as the cost and some hydraulic constraints. As a result, continuous diameters are obtained and rounded to the nearest commercial diameter in order to satisfy all the hydraulic constraints. This methodology was evaluated over a network with 896 manholes, with a reduced diameter range from 0.2 to 0.4 meters ( $m$ ), and additional simplifications. The most popular method, used by numerous authors, is genetic algorithms (GAs). Haghghi

& Bakhshipour (2012) and Palumbo *et al.* (2014) used GAs to design urban drainage networks, deciding the size and ground elevation of the pipes. Cimorelli *et al.* (2013) combined a GA with hydrologic and hydraulic models that evaluate the water depth and water discharge through the network. Cozzolino *et al.* (2015) used a GA combined with a steady and uniform flow hydraulic module to achieve a minimum-cost design of drainage networks. This methodology was tested over networks with nine and 37 pipes, conveying to a near-optimal solution in around 1,700 generations of the GA.

Another line of heuristic schemes combines GA with other approaches. Cisty (2010) proposed a hybrid between GA and LP; Pan & Kao (2009) mixed GA with quadratic programming; and Haghghi *et al.* (2011) used GA with integer programming applied to water distribution networks. Afshar (2012) developed a rebirthing GA that performs a first quick design using a standard GA, narrowing the search space around the best solutions, and then creating a new random population that is evaluated over the narrowed search space in order to find near-optimal solutions quickly. This methodology reduces the sensitivity of the GA to the number of individuals in its population. Cimorelli *et al.* (2014) used an enhanced genetic algorithm to find a near-optimal design of rural drainage networks while making a simultaneous evaluation of the design discharges that vary due to climatic and hydrologic characteristics. Sanchez *et al.* (2014) combined a cellular automata (CA) model of urban growth, the evolutionary algorithm non-dominated sorting genetic algorithm-II, and a hydraulic simulator (Storm Water Management Model, SWMM) to design the pipes for future expansions of an existing sewer network. This multi-objective algorithm provides a set of non-dominated solutions, and the one that has the cheapest design cost and produces no flooding is ultimately selected. Other heuristic methods such as ant colony optimization algorithm (Afshar 2010; Moeini & Afshar 2012), simulated annealing (Sousa *et al.* 2002; Yeh *et al.* 2011), and Tabu Search (Yeh *et al.* 2011; Haghghi & Bakhshipour 2015) have also been used in sewer systems design.

Additional problems, closely related to the sewer network design, have been addressed in the literature. Fu *et al.* (2010) used artificial neural networks to describe urban wastewater systems and to predict combined sewer overflows discharges and treatment effluent characteristics. Likewise,

Austin et al. (2014) proposed a CA model to simulate sewer flow as figurative water blocks with a finite volume. Additionally, Mortazavi-Naeini et al. (2015) developed efficient multi-objective ant colony optimization-I in order to solve operational and infrastructure problems of urban water resources. Cimorelli et al. (2015) used a GA approach to find a near-optimal location and size of detention tanks in urban drainage networks. All of these aspects are related and relevant for sewer systems design, and are suitable for further extensions in more complex and broad designs.

In summary, available exact methods for the design problem often require multiple assumptions and the simplification of components. Additionally, they do not always scale for large instances. Heuristic approaches are more flexible and scalable, but do not offer an optimal (exact) solution. This paper proposes an exact, exhaustive, and flexible framework to find the cost-optimal design of a series of sewer pipes using a DP-based optimization engine. The design problem is modeled as a shortest path problem (cf., Bellman 1958; Ahuja et al. 1993), where the underlying graph considers every feasible combination of diameter and slope for each pipe in the series. Thus, a shortest path on the graph encodes an optimal hydraulic design decision. Hydraulic aspects that guarantee the proper operation of the system are taken into account in the construction of the underlying graph, and additional side constraints of the system can be addressed in the proposed methodology. Finally, different cost functions can be flexibly modeled in order to accurately estimate the construction cost according to the country where the sewer network is located.

The rest of the paper is organized as follows: the following section presents the proposed methodology. Then, the solution strategy is explained in detail. Finally, several numerical experiments over theoretical series of pipes are presented, followed by the last section that concludes the paper and outlines future research.

## METHODOLOGY

### The shortest path problem

The *shortest path problem* is defined over a graph  $G = (\mathcal{N}, \mathcal{A})$ , where  $\mathcal{N} = \{v_0, \dots, v_n\}$  is a set of

nodes,  $\mathcal{A} = \{(v_i, v_j) | v_i \in \mathcal{N}, v_j \in \mathcal{N}, v_i \neq v_j\}$  is a set of arcs, and  $c_{ij}$  is the cost (e.g., distance, time, etc.) of traversing arc  $(v_i, v_j) \in \mathcal{A}$ . The objective is to find a minimum-cost path (e.g., shortest distance or minimum travel time) from a specific initial node  $v_s \in \mathcal{N}$  to a destination node  $v_t \in \mathcal{N}$  (Bellman 1958). Main applications of this optimization model arise in transportation and routing problems. Formally, the shortest path problem is defined as follows:

$$\min \sum_{(v_i, v_j) \in \mathcal{A}} c_{ij} x_{ij} \quad (1)$$

s.t.

$$\sum_{\{j | (v_i, v_j) \in \mathcal{A}\}} x_{ij} - \sum_{\{j | (v_j, v_i) \in \mathcal{A}\}} x_{ji} = \begin{cases} 1 & v_i = v_s \\ 0 & v_i \in \mathcal{N} \setminus \{v_s, v_t\} \\ -1 & v_i = v_t \end{cases}; \quad (2)$$

$$x_{ij} \in \{0, 1\} \quad \forall v_i \in \mathcal{N}, v_j \in \mathcal{N}; \quad (3)$$

where  $x_{ij}$  is a binary variable valued as one if the arc  $(v_i, v_j) \in \mathcal{A}$  belongs to the shortest path and valued at zero otherwise. The objective function (1) minimizes the costs of the path. This is subject to Equation (2) that establishes the flow balance constraints that guarantee the structural properties of a path, and Equation (3) that restricts the variables to be binary-valued.

To solve this problem, there are different algorithms such as Bellman–Ford algorithm (Algorithm 1) or Dijkstra’s algorithm that use labels (DP) to find an optimal solution (Ahuja et al. 1993). In this work, the Bellman–Ford algorithm is used due to the particular structure of the underlying graph, i.e., a directed-acyclic graph that is ordered in a topological fashion ( $i < j$  for every arc  $(v_i, v_j) \in \mathcal{A}$ ). This label-correcting method starts setting a cumulative cost (label)  $V(v_s) = 0$  for the initial node  $v_s \in \mathcal{N}$  and  $V(v_i) = \infty$  for the remaining nodes of the graph. Then, each node  $v_i \in \mathcal{N}$  is evaluated by scanning every outgoing arc  $(v_i, v_j) \in \mathcal{A}$ . If  $V(v_j) > V(v_i) + c_{ij}$ , then the label for  $v_j \in \mathcal{N}$  is updated with the new minimum cumulative cost  $V(v_j) = V(v_i) + c_{ij}$  and its parent (predecessor) node  $p(v_j) = v_i$ . When all the nodes of the graph are evaluated, the cost labels are optimal (Ahuja et al. 1993).

**Input:**  $\mathcal{N}$ , set of nodes;  $\mathcal{A}$ , set of arcs.

**Output:** void

```

1:  $V(v_0) = 0$ 
2:  $p(v_0) = \text{null}$ 
3: for  $v_i \in \mathcal{N} | i \neq 0$  do
4:    $V(v_i) = \infty$ 
5: end for
6: for  $v_i \in \mathcal{N}$  do
7:   for  $v_j \in \mathcal{N} | (v_i, v_j) \in \mathcal{A}$  do
8:     if  $V(v_j) > V(v_i) + c_{ij}$  then
9:        $V(v_j) \leftarrow V(v_i) + c_{ij}$ 
10:       $p(v_j) \leftarrow v_i$ 
11:     end if
12:   end for
13: end for

```

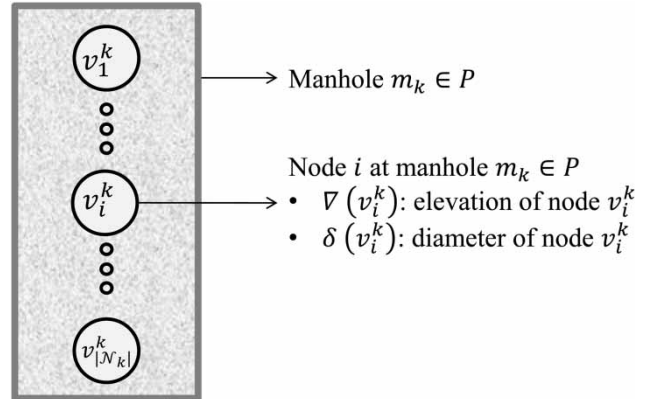
**Algorithm 1** | Bellman-Ford algorithm.

## Graph modeling

To build the underlying graph of the shortest path problem, several parameters are utilized as input data. Let  $P = \{m_0, m_1, \dots, m_K\}$  be the set of manholes comprised in the series of pipes, where manhole  $m_0 \in P$  is the starting point, manhole  $m_K \in P$  is the discharge point of the sewer system, and  $Q_k$  and  $\nabla_k$  are the inflow and ground elevation at manhole  $m_k \in P$ , respectively. Due to the sequential structure of the sewer system, the design flow rate for a pipe between manholes  $m_k$  and  $m_{k+1}$  is precomputed as  $Q_d(k, k+1) = \sum_{i=0}^k Q_i$ . Additionally, let  $D = \{d_1, d_2, \dots, d_d\}$  be the set of commercial diameters available and  $l$  be the length of each pipe in the series.

In the underlying graph that models the series of pipes, each manhole is represented by a subset of nodes and each arc connects two nodes of consecutive manholes to represent a pipe with a particular diameter and slope. Formally,  $\mathcal{N} = \{v_0, v_1, \dots, v_n\}$  is the set of all nodes in the graph, and  $\mathcal{N}_k = \{v_1^k, v_2^k, \dots, v_{|\mathcal{N}_k}^k\}$  is the subset of nodes ( $\mathcal{N}_k \subset \mathcal{N}$ ) arranged at manhole  $m_k \in P$ . Each node in  $\mathcal{N}_k$  comprises two attributes: an elevation  $\nabla(v_i^k)$ , which represents the invert elevation of the pipe at manhole  $m_k \in P$ , and a commercial diameter  $\delta(v_i^k)$ , where  $\delta(v_i^k) \in D$  represents the diameter of an incoming pipe to manhole  $m_k \in P$  that starts at manhole  $m_{k-1} \in P$ . Figure 1 shows an example of the subset  $\mathcal{N}_k$  for a given manhole  $m_k \in P$ .

Additionally, there is a set of arcs  $\mathcal{A} = \{(v_i^k, v_j^{k+1}) | v_i^k \in \mathcal{N}_k, v_j^{k+1} \in \mathcal{N}_{k+1}\}$  that represents the pipes between two consecutive manholes. Each arc is defined between nodes  $v_i^k \in \mathcal{N}_k$  and  $v_j^{k+1} \in \mathcal{N}_{k+1}$ , where



**Figure 1** | Subset of nodes that belong to manhole  $m_k \in P$ .

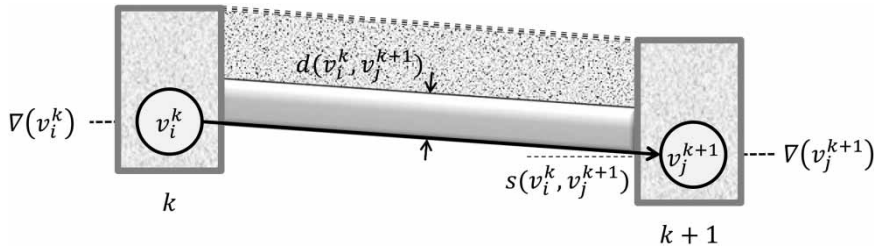
$j$ th is the  $i$ th node at manhole  $m_k \in P$  and  $v_j^{k+1}$  is the  $j$ th node of the immediately following manhole  $m_{k+1} \in P$ . Each arc has a cost attribute  $c(v_i^k, v_j^{k+1})$ , which represents the total construction costs of the pipe. The arc cost corresponds to the cost of the pipe plus the excavation cost. In our study, the cost function established by Equation (4) is the one used for sewer systems in Colombia:

$$C = \frac{1.32}{2,000} (9,579.31d^{0.5737} + 1,163.77V^{1.31}), \quad (4)$$

where  $d$  is the pipe diameter and  $V$  is the excavation volume required for placing the pipe (Navarro 2009). It is worth noting that without affecting the methodology, this equation can be replaced by any other equation or value that better fits the construction costs of the country in which the system is going to be constructed. Also, the equation could include the costs of manholes and pumping stations, as proposed by Marchionni et al. (2014).

Given that the information of the diameter is saved as an attribute of the nodes, Equation (5) states that the diameter associated with arc  $(v_i^k, v_j^{k+1}) \in \mathcal{A}$  takes the value of the head node  $v_j^{k+1} \in \mathcal{N}_{k+1}$ . Likewise, Equation (6) describes the slope associated with arc  $(v_i^k, v_j^{k+1}) \in \mathcal{A}$  as a function of the elevation of the corresponding nodes. Figure 2 shows a representation of a pipe as an arc  $(v_i^k, v_j^{k+1}) \in \mathcal{A}$  and its corresponding attributes.

$$d(v_i^k, v_j^{k+1}) = \delta(v_j^{k+1}) \quad (5)$$



**Figure 2** | Representation of a pipe in a sewer system as an arc.

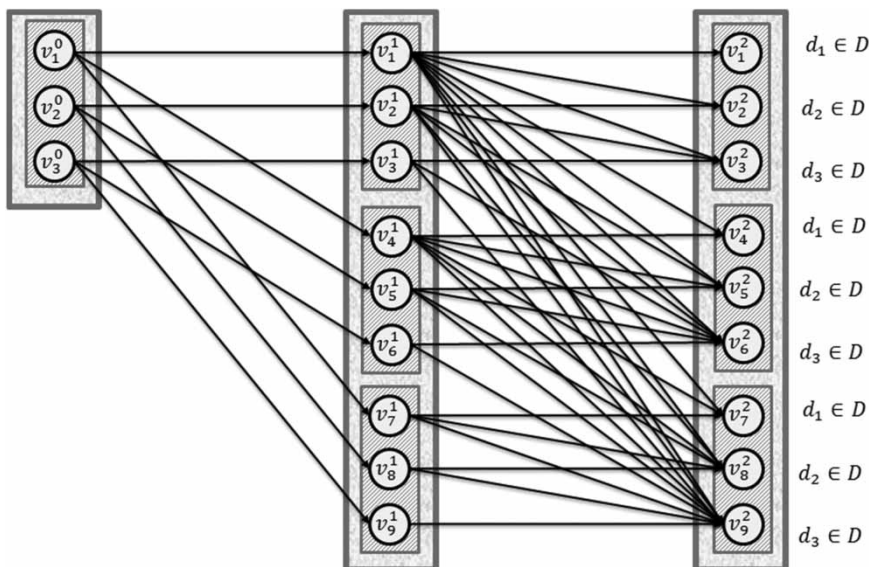
$$s(v_i^k, v_j^{k+1}) = \frac{\nabla(v_i^k) - \nabla(v_j^{k+1})}{l} \quad (6)$$

Figure 3 shows an illustrative example of a graph that represents a sewer system with three manholes ( $m_0, m_1, m_2$ ) and three different commercial diameters (i.e.,  $D = \{d_1, d_2, d_3\}$ ). At each manhole, nodes are grouped in smaller rectangles to represent different depths at which the pipes can be placed, while the nodes within the groups represent different pipe diameters in an increasing order. For instance, nodes  $v_1^1, v_2^1$ , and  $v_3^1$  satisfy that  $\nabla(v_1^1) = \nabla(v_2^1) = \nabla(v_3^1)$  and  $\delta(v_1^1) = d_1 < \delta(v_2^1) = d_2 < \delta(v_3^1) = d_3$ . Nodes  $v_4^1, v_5^1$ , and  $v_6^1$  are placed  $\tilde{\nabla}$  units below the previous group of nodes and represent the same three commercial diameters, i.e.,  $\nabla(v_4^1) = \nabla(v_5^1) = \nabla(v_6^1) = \nabla(v_1^1) - \tilde{\nabla}$  and  $\delta(v_4^1) = d_1 < \delta(v_5^1) = d_2 < \delta(v_6^1) = d_3$ . The elevation change

$\tilde{\nabla}$  is an input parameter of the methodology that establishes a numeric tolerance of the designs.

Basic hydraulic constraints are taken into account as the graph is built. Feasible arcs between nodes must satisfy that  $\delta(v_i^k) \leq \delta(v_j^{k+1})$  to prevent obstructions in the system. Additionally, each arc  $(v_i^k, v_j^{k+1}) \in \mathcal{A}$  satisfies that  $\nabla(v_i^k) > \nabla(v_j^{k+1})$  to avoid adverse slopes. For manhole  $m_0$  there is only one possible placement at the upper excavation limit. Also, for the first pipe of the series the diameter of the upstream node must match the diameter of the downstream node.

It is noteworthy that the solution space of the problem is very large due to the exponential growth of design alternatives (i.e., hydraulically feasible paths from the first to the last manhole). However, state-of-the-art algorithms that solve the shortest path problem manage networks with several millions of nodes and arcs (Demetrescu



**Figure 3** | Graph representing a sewer system with two pipes in series for three commercial diameters and three possible invert elevations.

et al. 2006). Moreover, due to the particular acyclic-forward-directed structure of the graph, a specialized version of the shortest path algorithm discussed before is used to accelerate the solution strategy. This specialized algorithm is discussed in detail in the solution strategy section.

After solving the shortest path problem, the solution obtained is a minimum-cost path that encodes the design decisions of the complete series. Notice that since the solution is a path, the methodology selects a single pipe (arc) for each link among all possible alternatives. By associating this graph representation with the mathematical formulation (Equations (1)–(3)), the objective function can be rewritten to detail the cost of a path (design). Equation (7) establishes the cost of an arc using the graph representation, while Equation (8) use this cost definition in the objective function.

$$c(v_i^k, v_j^{k+1}) = \frac{1.32}{2,000} \left( 9,579.31d(v_i^k, v_j^{k+1})^{0.5737} + 1,163.77V^{1.31} \right) \quad (7)$$

$$\min \sum_{k=0}^{K-1} \sum_{v_i^k \in \mathcal{N}_k} \sum_{v_j^{k+1} \in \mathcal{N}_{k+1}} c(v_i^k, v_j^{k+1}) x_{v_i^k, v_j^{k+1}} \quad (8)$$

Notice that Equation (7) depends on the pipe diameter  $d(v_i^k, v_j^{k+1})$  and on the excavation volume  $V$ . The latter is computed using the elevation values  $\nabla(\cdot)$  of the corresponding nodes  $v_i^k$  and  $v_j^{k+1}$ . Pipes are joined by their invert elevation in order to avoid hydraulic jumps into the system

that might cause erosion of sewer materials and release gases (Butler & Davies 2011).

## Hydraulic constraints

The graph presented in Figure 3 takes into account basic hydraulic constraints that prohibit decreasing diameters and adverse slopes. However, there is also a set of hydraulic constraints that are usually considered in the literature (and legislation) of sewer systems design that must be addressed. In this work, the hydraulic design uses the Darcy–Weisbach resistance equation (Brown 2002); nonetheless, the methodology could use any other resistance equation such as Manning's equation.

Hydraulic constraints ensure a proper operation of the sewer networks. They establish bounds for the constraints exhibited in Table 1. The minimum allowed diameter prevents obstructions in the system. The maximum filling ratio precludes surcharge problems and ensures ventilation in the system to avoid environmental difficulties. The minimum wall shear stress and velocity ensure a cleaning process inside the pipes; while the maximum velocity prevents problems such as erosion, cavitation, air entrapment, hydraulic jumps, among others. The minimum and maximum slope constraints are limited by the minimum and maximum velocities, respectively.

Finally, there is a constraint that bounds the depth at which pipes can be placed. These allowed depths are measured from the ground to the crown elevation of the pipes. The minimum depth ( $h_{min}$ ) protects the structure of

**Table 1** | Hydraulic design constraints

Constraint	Value	Condition
1 Minimum diameter	200 mm	Always
2 Maximum filling ratio	0.7 0.8 0.85	$d \leq 600$ cm $0.7 \leq Fr \leq 1.5$ Other cases
3 Minimum wall shear stress	2 Pa	$d \geq 450$ cm
4 Minimum velocity	0.75 m/s	$d < 450$ cm
5 Maximum velocity	5 m/s 10 m/s	$k_s > 0.0001$ $k_s < 0.0001$
6 Minimum slope	The one for which the minimum velocity and shear stress are obtained	
7 Maximum slope	The one for which the maximum velocity is obtained	

**Table 2** | Excavation limits

Road classification	$h_{min}$ (m)	$h_{max}$ (m)
Pedestrian or green area	0.75	5.0
Vehicular	1.2	5.0

the pipes and ensures that domestic discharges drain by gravity. The maximum depth ( $h_{max}$ ) limits axial loads over the pipes. These excavation bounds are presented in Table 2, according to RAS (2000).

### Solution strategy

This section presents the solution strategy as a three-step procedure based on the graph modeling and the shortest path algorithm presented above. The first step consists in generating the nodes for each manhole. In the second step, the graph that represents all the possible solutions is built with only feasible arcs while Bellman–Ford algorithm is executed simultaneously. The third step extracts the optimal solution from the graph.

Given the input data described in the graph modeling section, the first step is to generate all the nodes of the graph and organize them in subsets at every manhole  $m_k \in P$ . Algorithm 2 presents the nodes generation process

for every manhole  $m_k \in P$ , which receives as input parameters the set of commercial diameters  $D$ , the set of manholes  $P$ , and the ground elevation  $\nabla$  of each manhole. Lines 1–19 generate the nodes for each manhole  $m_k \in P$ . Lines 2 and 3 set the upper and lower excavation bounds. Lines 4–8 create the nodes for the first manhole  $m_0$  (one node per diameter in  $D$ ), and are placed at the upper limit. For the other manholes, the node generation process is presented in lines 9–18. In these manholes there is one node for each combination of diameter  $\delta(v_i^k)$  and elevation  $\nabla(v_i^k)$ . Line 10 shows the propagation over all diameters in  $D$  and line 11 sets the starting elevation  $\nabla_i^k$  to the first feasible invert elevation at which a pipe could reach manhole  $m_k \in P$ . This process continues until  $\nabla_i^k$  reaches the lower limit ( $h_{max}$  below the ground level), which is evaluated in line 12. This procedure generates as many nodes at each elevation as available diameters. The number of evaluated elevations depends on the elevation change used ( $\tilde{\nabla}$ ). For example, if  $\tilde{\nabla} = 1$  cm and the pipe is placed below a vehicular road (see Table 2), there are 380 elevations to evaluate in a range of 3.8 m (from 1.2 m to 5 m above the ground level). Similarly, if  $\tilde{\nabla} = 10$  cm, there are 38 elevations to evaluate. Therefore, there is a tradeoff between the precision of the design versus the computational effort to create the graph and solve the corresponding shortest path problem.

**Input:**  $\mathcal{D}$ , set of diameters;  $\mathcal{P}$ , set of manholes;  $\nabla$ , set of manholes' ground elevations;  $\tilde{\nabla}$ , elevation change.

**Output:** void

```

1: for  $m_k \in P$  do
2:    $\nabla_{upper} = \nabla_k - h_{min}$ 
3:    $\nabla_{lower} = \nabla_k - h_{max}$ 
4:   if  $k = 0$  then
5:     for  $d \in \mathcal{D}$  do
6:        $v_i^0 = \text{new\_node}(d, \nabla_{upper} - d)$ 
7:        $\mathcal{N}_0 \leftarrow \mathcal{N}_0 \cup \{v_i^0\}$ 
8:     end for
9:   else
10:    for  $d \in \mathcal{D}$  do
11:       $\nabla_i^k = \nabla_{upper} - d$ 
12:      while  $\nabla_i^k \geq \nabla_{lower} - d$  do
13:         $v_i^k = \text{new\_node}(d, \nabla_i^k)$ 
14:         $\mathcal{N}_k \leftarrow \mathcal{N}_k \cup \{v_i^k\}$ 
15:         $\nabla_i^k = \nabla_i^k - \tilde{\nabla}$ 
16:      end while
17:    end for
18:  end if
19: end for

```

**Algorithm 2** | Create nodes for each manhole.

**Input:**  $\mathcal{N}$ , set of nodes;  $l$ , pipe length;  $Q_d(k, k+1)$ , pipe design flow.

**Output:** void

```

1: for  $m_k \in P$  do
2:   for  $v_i^k \in \mathcal{N}_k$  do
3:     for  $v_j^{k+1} \in \mathcal{N}_{k+1}$  do
4:       if  $\delta(v_j^{k+1}) \geq \delta(v_i^k)$  then
5:          $s_{ij} \leftarrow (\nabla(v_i^k) - \nabla(v_j^{k+1}))/l$ 
6:         if  $s_{ij} > 0$  then
7:            $c_{ij} \leftarrow \text{calculate\_cost}(\delta(v_j^{k+1}), \nabla(v_i^k), \nabla(v_j^{k+1}), l)$ 
8:           if  $V(v_j^{k+1}) > V(v_i^k) + c_{ij}$  then
9:              $Q \leftarrow \text{calculate\_hydraulics}(\delta(v_j^{k+1}), s_{ij})$ 
10:            if  $Q \geq Q_d(k, k+1)$  then
11:              Calculate_normal_flow_depth( $\delta(v_j^{k+1}), Q_d(k, k+1), s_{ij}$ )
12:              if validate_constraints( $\delta(v_j^{k+1})$ ) then
13:                 $V(v_j^{k+1}) \leftarrow V(v_i^k) + c_{ij}$ 
14:                 $p(v_j^{k+1}) \leftarrow v_i^k$ 
15:              end if
16:            end if
17:          end if
18:        end if
19:      end if
20:    end for
21:  end for
22: end for

```

**Algorithm 3** | Build the graph and execute Bellman-Ford algorithm.

The next step is to create only those arcs that satisfy all the commercial and hydraulic constraints presented in Table 1. Algorithm 3 presents the procedure that builds the graph and solve the shortest path problem. The process evaluates all arcs between two consecutive manholes as described in lines 1–3. First, line 4 guarantees that the diameter of the downstream node  $\delta(v_j^{k+1})$  is always greater than or equal to the diameter of the upstream node  $\delta(v_i^k)$  to avoid obstructions. Line 5 calculates the slope at which the pipe will be placed in the case of selecting the current arc as part of the optimal path. Line 6 evaluates if the slope is positive, i.e., in favor of gravity. Line 7 calculates the arc cost following Equation (7); where the pipe diameter is set as in Equation (5) and the excavation volume is calculated according to the elevations of the nodes and the length of the pipe. While building the graph, the Bellman-Ford algorithm is simultaneously executed in order to find the shortest path. Line 8 applies the label correcting algorithm, however, before updating the cumulative cost and parent labels (see lines 13 and 14), the hydraulic constraints must be validated. Line 9, computes the flow rate  $Q$  for the pipe represented by the arc  $(v_i^k, v_j^{k+1}) \in \mathcal{A}$  (see Algorithm 4). Line 10 verifies that the computed flow rate is at least the

design flow rate of the current link ( $Q_d(k, k+1)$ ). Line 11 calculates the normal flow depth  $y_n$  required to (exactly) meet the design flow rate  $Q_d(k, k+1)$  of the link (see Algorithm 5). Finally, the label correcting process continues as shown in lines 12–14 if the arc  $(v_i^k, v_j^{k+1}) \in \mathcal{A}$  satisfies all the hydraulic constraints presented in Table 1 and validated in Algorithm 6. Line 13 sets the new minimum cumulative cost of node  $v_j^{k+1} \in \mathcal{N}_{k+1}$  and line 14 saves node  $v_i^k \in \mathcal{N}_k$  as the parent node of the former.

**Input:**  $d_{ij}$ , pipe diameter;  $s_{ij}$ , pipe slope.

**Output:**  $Q$ , maximum flow rate in the pipe.

```

1:  $y_{max} \leftarrow 0.85d_{ij}$ 
2: if  $d_{ij} \leq 0.6$  then
3:    $y_{max} \leftarrow 0.7d_{ij}$ 
4: end if
5:  $\theta \leftarrow \pi + 2 \sin^{-1} \left( \frac{y_{max} + d_{ij}/2}{d_{ij}/2} \right)$ 
6:  $A \leftarrow \frac{1}{8} (\theta - \sin \theta) d_{ij}^2$ 
7:  $P \leftarrow \frac{1}{2} \theta d_{ij}$ 
8:  $R \leftarrow \frac{1}{4} \left( 1 - \frac{\sin \theta}{\theta} \right) d_{ij}$ 
9:  $\tau_0 \leftarrow \rho g R s_{ij}$ 
10:  $v \leftarrow -2 \sqrt{8gR s_{ij}} \log_{10} \left( \frac{k_s}{14.8R} + \frac{2.51\nu}{4R \sqrt{8gR s_{ij}}} \right)$ 
11:  $Q \leftarrow v/A$ 
12: return  $Q$ 

```

**Algorithm 4** | Calculate hydraulics.



**Input:**  $d_{ij}$ , pipe diameter;  $s_{ij}$ , pipe slope;  $Q_d(k, k + 1)$ , pipe design flow.

**Output:** Void

```

1:  $y_1 \leftarrow 0$ 
2:  $y_2 \leftarrow y_{max}$ 
3:  $y_n \leftarrow \frac{y_1 + y_2}{2}$ 
4: while  $y_1 - y_2 > 1 \times 10^{-6}$  do
5:    $\theta \leftarrow \pi + 2 \sin^{-1} \left( \frac{y_n + d_{ij}/2}{d_{ij}/2} \right)$ 
6:    $A \leftarrow \frac{1}{8} (\theta - \sin \theta) d_{ij}^2$ 
7:    $P \leftarrow \frac{1}{2} \theta d_{ij}$ 
8:    $R \leftarrow \frac{1}{4} \left( 1 - \frac{\sin \theta}{\theta} \right) d_{ij}$ 
9:    $\tau_0 \leftarrow \rho g R s_{ij}$ 
10:   $v \leftarrow -2 \sqrt{8g R s_{ij}} \log_{10} \left( \frac{k_s}{14.8R} + \frac{2.51\nu}{4R\sqrt{8g R s_{ij}}} \right)$ 
11:   $Q \leftarrow v/A$ 
12:  if  $Q \leq Q_d(k, k + 1)$  then
13:     $y_1 \leftarrow y_n$ 
14:  else
15:     $y_2 \leftarrow y_n$ 
16:  end if
17:   $y_n \leftarrow \frac{y_1 + y_2}{2}$ 
18: end while

```

**Algorithm 5** | Calculate normal flow depth.

Note that the cumulative weight of all the hydraulic constraints validated in Algorithm 3 ultimately defines whether the arc  $(v_i^k, v_j^{k+1}) \in \mathcal{A}$  is feasible or not. Additionally, the optimal path (optimal design) only considers feasible arcs. Notice also that the algorithm applies the Bellman–Ford verification at line 8, before the hydraulic verifications. Since the arc cost is much easier to compute than the validation of hydraulic feasibility, it is worth verifying first the labels of the shortest path algorithm because it might save several hydraulic verifications of suboptimal solutions.

Algorithm 4 presents the method that calculates the flow rate  $Q$  transported through the arc  $(v_i^k, v_j^{k+1}) \in \mathcal{A}$ .

**Input:**  $d_{ij}$ , pipe diameter

**Output:** boolean

```

1: feasible_arc  $\leftarrow$  True
2: if  $k_s > 0.0001$  and  $v > 5$  then
3:   feasible_arc  $\leftarrow$  False
4: else if  $k_s < 0.0001$  and  $v > 10$  then
5:   feasible_arc  $\leftarrow$  False
6: else if  $\tau_0 \leq 2$  and  $d_{ij} \leq 0.45$  then
7:   feasible_arc  $\leftarrow$  False
8: else if  $v \leq 0.75$  and  $d_{ij} \leq 0.45$  then
9:   feasible_arc  $\leftarrow$  False
10: else if  $\frac{y_n}{d_{ij}} \geq 0.8$  and  $0.7 \leq Fr \leq 1.5$  then
11:   feasible_arc  $\leftarrow$  False
12: end if
13: return feasible_arc

```

**Algorithm 6** | Validate constraints.

Lines 1–4 set the maximum flow depth  $y_{max}$  according to the maximum filling ratio (85% or 70% depending on the pipe diameter). Lines 5–8 calculate the geometric properties that describe a pipe with partial flow: central angle  $\theta$ , flow area  $A$ , wetted perimeter  $P$ , and hydraulic radius  $R$ . Finally, lines 9–11 calculate the following hydraulic properties: wall shear stress, velocity, and flow rate. In this case, the flow rate represents the maximum capacity of the pipe.

After verifying that the maximum flow is greater than the design flow rate  $Q_d(k, k + 1)$  in Algorithm 3 (line 10), the exact normal flow depth is calculated in order to transport exactly the design flow rate. Algorithm 5 presents a binary search procedure to find the exact normal flow depth of a particular pipe with a tolerance of  $10^{-6}m$ . Line 1 sets the lower bound for the flow depth to 0, and line 2 sets the upper bound to  $y_{max}$ . Line 3 sets the new normal flow depth  $y_n$  as the average between both bounds. Lines 4–18 define the binary search with the corresponding tolerance. In lines 5–11, all the geometric and hydraulic properties are recalculated with the new normal flow depth  $y_n$ . Lines 12–16 set a new search range for the normal flow depth that is adjusted according to the calculated flow rate  $Q$  and the design flow rate  $Q_d(k, k + 1)$ . Finally, line 17 sets a new normal flow depth  $y_n$  that is either used in the next step of the search or accepted by the algorithm if the tolerance is met.

Once the normal depth  $y_n$  and all the hydraulic properties are updated, the hydraulic constraints established in Table 1 must be verified, as shown in Algorithm 6. A Boolean flag is used in line 1 to verify that a pipe satisfies all the hydraulic constraints. If there is at least one constraint that is not satisfied, the arc that represents that pipe is considered infeasible and is discarded for the design. Lines 2 and 4 verify the maximum velocity for rough and smooth materials, respectively.

---

**Input:**  $\mathcal{N}$ , set of nodes.  
**Output:**  $\mathcal{P}$ , optimal path.

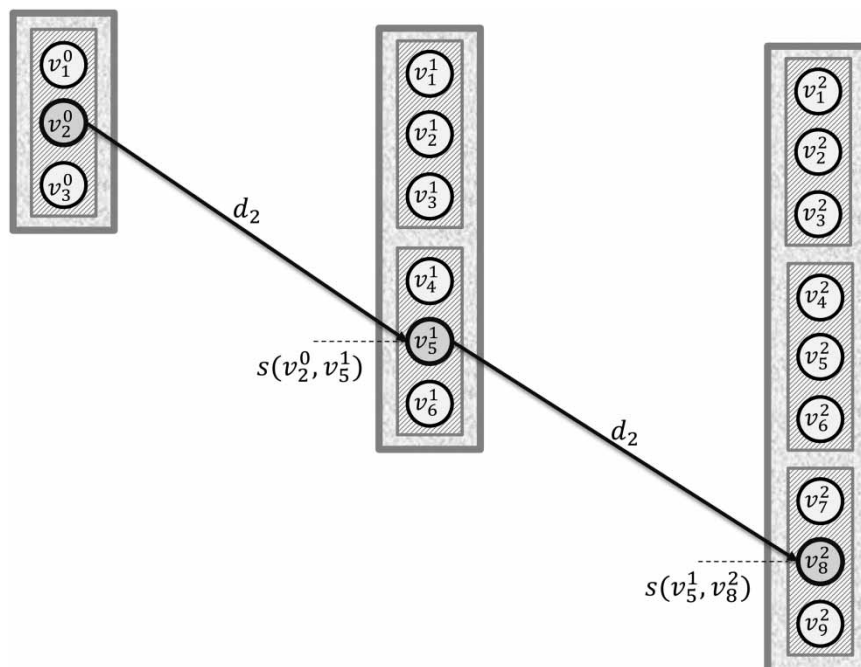
```

1: min_cost  $\leftarrow \infty$ 
2: min_cost_node  $\leftarrow \text{null}$ 
3: for  $v_i \in \mathcal{N}_K$  do
4:   if  $V(v_i) < \text{min\_cost}$  then
5:     min_cost  $\leftarrow V(v_i)$ 
6:     min_cost_node  $\leftarrow v_i$ 
7:   end if
8: end for
9:  $\mathcal{P} \leftarrow \{\}$ 
10: parent_node  $\leftarrow \text{min\_cost\_node}$ 
11: while parent_node  $\neq \text{null}$  do
12:    $\mathcal{P} \leftarrow \mathcal{P} \cup \{\text{parent\_node}\}$ 
13:   parent_node  $\leftarrow p(\text{parent\_node})$ 
14: end while
15: return  $\mathcal{P}$ 

```

---

**Algorithm 7** | Get optimal solution.



**Figure 4** | Design solution of two pipes in series of a sewer network.

The minimum wall shear stress and minimum velocity are evaluated in lines 6 and 8. Finally, line 10 verifies that the filling ratio is below 80% when having critical flow.

Finally, the graph that includes all feasible arcs is explored in a backwards manner, in order to extract the solution as shown in Algorithm 7. Lines 1–8 search over all the nodes at the last manhole ( $v_i^K \in \mathcal{N}_K$ ) in order to find the minimum cumulative cost node that stands for the minimum-cost alternative. Lines 9–15 recover the complete path by finding the parent nodes that belong to the shortest path  $\mathcal{P}$ .

Figure 4 shows the shortest path of the example presented in Figure 3. Path  $\mathcal{P} = \{v_2^0, v_5^1, v_8^2\}$  encodes a design in which both pipes use the second commercial diameter ( $d_2 \in D$ ) and the slopes of the pipes are in favor of gravity.

## NUMERICAL EXAMPLES

### Experiment setup

The algorithm is tested over 16 theoretical instances that are generated varying the number of pipes in the series (5, 10, 15, and 20), using two different materials (concrete and

PVC) and two different topographies (steep and flat ground). The algorithm is coded in Java and compiled using Eclipse SDK version 4.2.1. The experiments are performed on a computer with an Intel Core i7-3770 CPU @3.4 GHz with 8GB of RAM allocated to the memory heap size of the Java Virtual Machine on Windows 7. Computational time to solve the design problem measures the three steps of the proposed methodology, and this performance metric is reported individually for each instance.

Table 3 presents the inflow  $Q_k$  and ground elevation  $\nabla_k$  for each of the 21 manholes conforming the series of pipes (including manhole  $m_0$ ). Table 4 presents the design flow rate  $Q_d$ , the length  $l$ , and the corresponding upstream ( $k$ ) and downstream ( $k + 1$ ) manholes for each of the 20 pipes of the series. The first six, 11, and 16 manholes of the dataset are used to build smaller experiments with five, 10, and 15 pipes, respectively.

Table 3 | Manhole input data

Manhole	$Q_k$ [ $m^3/s$ ]	$\nabla_k$	
		Steep ground [m]	Flat ground [m]
0	0.0246	161.5	160.6
1	0.0114	160.5	160.7
2	0.0170	160.2	160.7
3	0.0230	160.2	160.7
4	0.0145	160.2	160.6
5	0.0125	160.2	160.5
6	0.0336	160.2	160.5
7	0.0332	159.9	160.7
8	0.0692	159.8	160.7
9	0.0246	159.8	160.9
10	0.0582	159.6	160.8
11	0.1552	159.6	160.7
12	0.0372	159.6	160.7
13	0.0283	159.2	160.7
14	0.0506	158.3	160.8
15	0.0360	158.2	160.6
16	0.0546	158.0	160.8
17	0.0460	157.9	160.6
18	0.1190	157.9	160.7
19	0.0215	157.6	160.8
20	–	157.6	160.7

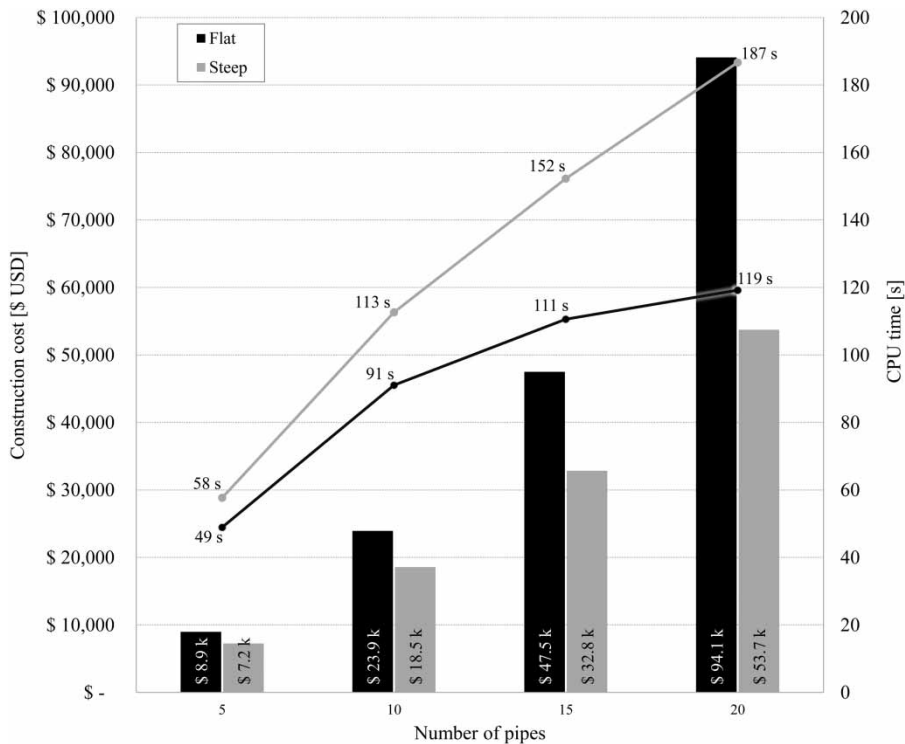
## Numerical results

Figure 5 summarizes the results for the design of various different series of pipes using PVC pipes and an elevation change of  $\tilde{\nabla} = 1$  cm. The figure shows how the cost (bars) and the computational time (lines) grow as the size of the series of pipes increases for both types of topographical setups.

Costs achieved in the steep topography are below the costs obtained in the flat topography. This is expected because in a steep topography, ground slopes can reduce the excavation volume required to place the pipes. In a flat topography, instead, pipes always require deeper excavation or increased diameter with every new link. The opposite occurs with the computational time, in which flat topographies are solved faster than the steep ones, and computational time does not grow exponentially, unlike the number of pipes in the series which increases. This

Table 4 | Pipe input data

Pipe	$Q_d$ [ $m^3/s$ ]	$l$ [m]	Manhole	
			$k$ [-]	$k + 1$ [-]
1	0.0246	180	1	2
2	0.0360	121	2	3
3	0.0530	205	3	4
4	0.0760	200	4	5
5	0.0905	123	5	6
6	0.1030	142	6	7
7	0.1366	115	7	8
8	0.1698	135	8	9
9	0.2390	168	9	10
10	0.2636	172	10	11
11	0.3218	120	11	12
12	0.4770	123	12	13
13	0.5142	136	13	14
14	0.5425	196	14	15
15	0.5931	115	15	16
16	0.6291	180	16	17
17	0.6837	121	17	18
18	0.7297	205	18	19
19	0.8487	195	19	20
20	0.8702	200	20	21



**Figure 5** | Construction cost and computational time for the designed series of pipes using PVC pipes and  $\tilde{v} = 1 \text{ cm}$ .

occurs because for a long series of pipes the solution space between the excavation limits (i.e., 3.8 m in this experiment) is very small, and therefore, thousands of feasible partial alternatives reach the lower excavation limit before reaching the outfall manhole.

Table 5 presents the design of a series of 20 pipes on flat ground using PVC pipes and  $\tilde{v} = 1 \text{ cm}$ . The downstream invert elevation of each pipe determines the upstream invert elevation for the next pipe. For the first pipe, the upstream invert elevation corresponds to the ground elevation minus the minimum depth ( $h_{min}$ ) and the pipe diameter. In this case, the invert elevation of the first pipe is 159.17 m.

Table 6 presents the design of another series of 20 pipes on steep ground with the same elevation change and material ( $\tilde{v} = 1 \text{ cm}$  and PVC pipes). In this case, the invert elevation for the first pipe is 160.1 m.

The effect of varying the elevation change  $\tilde{v}$  from 1 cm to 5 cm and 10 cm is also analyzed. The lowest cost is achieved with a highest precision ( $\tilde{v} = 1 \text{ cm}$ ) because more alternatives are evaluated. In this experiment, the

same costs are found using  $\tilde{v} = 1 \text{ cm}$  and  $\tilde{v} = 5 \text{ cm}$ , but for  $\tilde{v} = 10 \text{ cm}$  the cost increases in all the instances. Table 7 presents the relative gap between the cost of the solutions obtained with  $\tilde{v} = 1 \text{ cm}$  and  $\tilde{v} = 10 \text{ cm}$ . For example, the cost of a series of 20 pipes on steep ground using PVC pipes and  $\tilde{v} = 1 \text{ cm}$  is \$53,738.8, but using  $\tilde{v} = 10 \text{ cm}$  the cost rises to \$61,813.4. Thus, the relative gap is 15% and is due to the accuracy reached with each elevation change  $\tilde{v}$ .

For the flat ground setup, no feasible solutions are found for the series of 20 pipes using  $\tilde{v} = 10 \text{ cm}$  (neither for concrete pipes nor for PVC pipes). This occurs because there are fewer alternatives in the 3.8 m range above the ground to place the deeper pipes.

On the other hand, the computational time is far less for a design with an elevation change of  $\tilde{v} = 10 \text{ cm}$  than with an elevation change of  $\tilde{v} = 1 \text{ cm}$  or  $\tilde{v} = 5 \text{ cm}$ . Figure 6 presents the computational time and construction cost for each of the 24 numerical examples with PVC pipes. These examples correspond to the combination of two different topographies (flat ground and steep ground), four values

**Table 5** | Design of a series of 20 pipes on flat ground using PVC pipes and  $\tilde{\nabla} = 1 \text{ cm}$ 

Pipe	Invert elevation downstream [m]	s [-]	d [m]	$y_n/d$ [-]	v [m/s]	$\tau_o$ [Pa]	Fr [-]	Q [m <sup>3</sup> /s]	Cost [USD]
1	158.873	0.002	0.250	0.628	0.757	1.137	0.659	0.025	\$1,395.84
2	158.720	0.001	0.350	0.497	0.753	1.081	0.651	0.036	\$1,182.84
3	158.507	0.001	0.350	0.700	0.757	1.056	0.511	0.054	\$2,384.75
4	158.301	0.001	0.400	0.690	0.822	1.195	0.525	0.076	\$2,678.61
5	158.130	0.001	0.400	0.697	0.968	1.608	0.612	0.091	\$1,522.13
6	157.893	0.002	0.500	0.689	1.237	2.416	0.707	0.179	\$2,233.80
7	157.717	0.002	0.500	0.695	1.179	2.212	0.669	0.172	\$1,914.00
8	157.532	0.001	0.530	0.689	1.154	2.108	0.641	0.187	\$2,675.43
9	157.334	0.001	0.600	0.691	1.148	2.040	0.598	0.239	\$4,234.47
10	157.147	0.001	0.700	0.841	1.239	2.271	0.482	0.428	\$5,224.04
11	157.024	0.001	0.700	0.807	1.200	2.142	0.494	0.399	\$3,352.66
12	156.913	0.001	0.800	0.813	1.217	2.147	0.464	0.532	\$3,936.57
13	156.797	0.001	0.800	0.807	1.184	2.042	0.456	0.514	\$4,656.99
14	156.627	0.001	0.800	0.850	1.192	2.068	0.426	0.543	\$7,808.37
15	156.539	0.001	0.900	0.841	1.197	2.043	0.411	0.683	\$4,508.99
16	156.396	0.001	0.900	0.818	1.226	2.133	0.437	0.683	\$8,163.22
17	156.303	0.001	0.900	0.833	1.208	2.078	0.420	0.684	\$5,057.06
18	156.122	0.001	0.900	0.827	1.298	2.369	0.456	0.730	\$10,080.24
19	155.897	0.001	0.900	0.831	1.502	3.098	0.524	0.849	\$10,096.17
20	155.654	0.001	0.900	0.828	1.545	3.263	0.542	0.870	\$10,986.96
<b>Total</b>									<b>\$94,093.14</b>

for the number of pipes in the series (5, 10, 15, or 20), and three values for the elevation change  $\tilde{\nabla}$ . Computational time (y axis) is presented in a logarithmic scale, and the construction cost is exhibited in the x axis. Each point in the figure represents the solution of a particular setup.

This figure shows that a design that uses an elevation change of  $\tilde{\nabla} = 10 \text{ cm}$  is roughly 100 times faster than a design that uses an elevation change of  $\tilde{\nabla} = 1 \text{ cm}$ ; but a slightly higher cost is obtained. The former situation is due to exponential growth of the graph as the elevation change  $\tilde{\nabla}$  gets smaller. For instance, assuming five available diameters, the graph to solve the design problem encompasses 1,900 nodes per manhole with  $\tilde{\nabla} = 1 \text{ cm}$  (380 possible placements in 3.8 m by 5 available diameters), but only 190 nodes per manhole with  $\tilde{\nabla} = 10 \text{ cm}$  (38 possible placements in 3.8 m by 5 available diameters). However, a relative gap in costs might represent thousands of dollars. For example, the design of the series of 20 pipes with steep ground and

using PVC pipes takes 186.65 seconds using a  $\tilde{\nabla} = 1 \text{ cm}$  and 1.85 seconds using a  $\tilde{\nabla} = 10 \text{ cm}$ , but there is a relative gap of 15% in the cost of both solutions. Moreover, this gap increases proportional to the number of pipes. Therefore, it is worth spending more computational time minimizing the total cost as exhibited in Table 7, but it is also important to account for the computational time because it gives an insight into how the proposed methodology scales.

## CONCLUSIONS AND FUTURE WORK

In this work, a graph modeling framework is proposed in order to obtain the minimum-cost design of a series of pipes as a shortest path problem. The proposed graph takes into account every possible diameter and slope combination found in each of the series' pipes. A path on the graph

**Table 6** | Design of a series of 20 pipes on steep ground using PVC pipes and  $\tilde{\nabla} = 1 \text{ cm}$ 

Pipe	Invert elevation downstream [m]	s [-]	d [m]	$y_n/d$ [-]	v [m/s]	$\tau_o$ [Pa]	Fr [-]	Q [m <sup>3</sup> /s]	Cost [USD]
1	159.090	0.006	0.200	0.592	1.269	3.037	1.290	0.025	\$1,101.43
2	158.750	0.003	0.250	0.665	1.039	2.004	0.865	0.036	\$786.04
3	158.540	0.001	0.380	0.593	0.757	1.053	0.558	0.053	\$2,000.02
4	158.340	0.001	0.400	0.700	0.809	1.162	0.511	0.076	\$2,201.37
5	158.170	0.001	0.400	0.698	0.967	1.605	0.611	0.091	\$1,324.04
6	157.920	0.002	0.400	0.696	1.104	2.043	0.700	0.103	\$1,708.28
7	157.750	0.001	0.500	0.685	1.155	2.132	0.664	0.165	\$1,589.85
8	157.550	0.001	0.500	0.698	1.161	2.150	0.656	0.170	\$1,921.49
9	157.350	0.001	0.600	0.687	1.154	2.062	0.604	0.239	\$3,009.95
10	157.160	0.001	0.700	0.823	1.250	2.309	0.501	0.423	\$3,623.76
11	157.040	0.001	0.700	0.824	1.183	2.089	0.473	0.402	\$2,371.92
12	156.870	0.001	0.700	0.820	1.412	2.888	0.569	0.477	\$2,570.03
13	156.660	0.002	0.700	0.834	1.500	3.223	0.590	0.514	\$2,899.48
14	156.330	0.002	0.700	0.840	1.571	3.512	0.611	0.543	\$3,964.27
15	156.200	0.001	0.800	0.799	1.378	2.697	0.537	0.593	\$2,073.76
16	155.990	0.001	0.800	0.836	1.402	2.783	0.514	0.629	\$3,610.05
17	155.820	0.001	0.800	0.819	1.552	3.355	0.586	0.684	\$2,259.08
18	155.510	0.002	0.800	0.844	1.613	3.603	0.584	0.730	\$4,645.76
19	155.120	0.002	0.800	0.842	1.878	4.766	0.681	0.849	\$4,712.47
20	154.660	0.002	0.800	0.796	2.029	5.489	0.794	0.870	\$5,365.77
<b>Total</b>									<b>\$53,738.83</b>

additionally encodes a complete design. The graph modeling allows the representation of discrete commercial diameters, which avoids the use of continuous diameters that need to be rounded. The graph grows exponentially as the number of pipes increases, and this particular growth is taken advantage of by applying a highly efficient shortest path algorithm to the problem. Aside from the hydraulic and construction constraints considered in this work, this methodology allows the consideration of additional constraints by discarding arcs of the graph (for instance, a constraint related to the intersections with other utility networks).

The proposed methodology ensures the global optimal solution from an economic point of view because the graph considers all possible alternatives and the Bellman–Ford algorithm implicitly explores all of them. Without having to simplify hydraulic constraints, this methodology still obtains the optimal solution in a very short computational time using a standard desktop computer.

The elevation change  $\tilde{\nabla}$  used in the hydraulic design (1, 5, or 10 cm) affects both the solution and the computational time. With a small value for the elevation change  $\tilde{\nabla}$ , our methodology finds that the design with lower construction costs is the one that spends the most computational time. Therefore, there is a tradeoff between the solution quality and the computational time required to solve the problem. Additionally, the relative gap between the costs of those solutions obtained with different values of the elevation change  $\tilde{\nabla}$  are more significant as the number of pipes in the series increases. This is due to the fewer number of alternatives that are explored with larger values of  $\tilde{\nabla}$ . Finally, computational time is drastically reduced by first evaluating the economic feasibility according to the Bellman–Ford algorithm, and the hydraulic constraints afterwards. Latter evaluations are more expensive in terms of computational time, so it makes sense to verify the cost first in order to avoid hydraulic verifications of suboptimal solutions.

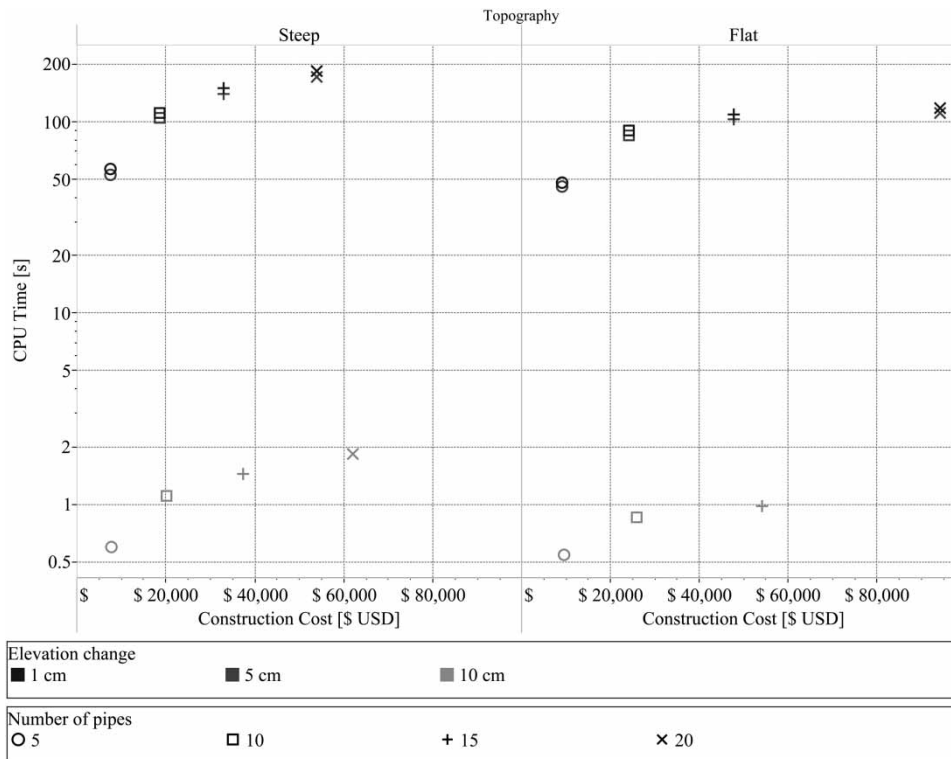


Figure 6 | Elevation change effect over the performance metrics.

Table 7 | Relative gap between the solutions obtained with  $\bar{v} = 1$  cm and  $\bar{v} = 10$  cm

Number of pipes	Relative gap			
	Steep ground		Flat ground	
	Concrete	PVC	Concrete	PVC
5	1%	4%	3%	3%
10	6%	8%	5%	8%
15	12%	13%	13%	13%
20	16%	15%	-	-

We suggest future work should extend this methodology for a complete sewer network; although several concepts of the methodology can already be extended in a straightforward fashion, there are plenty of new considerations that must be addressed. Layout decisions, such as the flow direction, definition of initial pipes, and intersections between two or more series in the network are some of the challenges that must be tackled to adapt the proposed methodology for complete sewer networks.

## REFERENCES

- Afshar, M. 2010 A parameter free continuous ant colony optimization algorithm for the optimal design of storm sewer networks: constrained and unconstrained approach. *Adv. Eng. Software* **41** (2), 188–195.
- Afshar, M. 2012 Rebirthing genetic algorithm for storm sewer network design. *Scientia Iranica* **19** (1), 11–19.
- Ahuja, R. K., Magnati, T. L. & Orlin, J. B. 1993 *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Upper Saddle River, NJ, USA.
- Austin, R., Chen, A., Savić, D. & Djordjević, S. 2014 Quick and accurate cellular automata sewer simulator. *J. Hydroinform.* **16** (6), 1359–1374.
- Bellman, R. 1958 On a routing problem. *Quart. Appl. Math.* **16** (1), 87–90.
- Brown, G. 2002 The history of the Darcy-Weisbach equation for pipe flow resistance. In: *Environmental and Water Resources History Sessions at ASCE Civil Engineering Conference and Exposition*. Washington, DC, USA, pp. 34–43.
- Butler, D. & Davies, J. W. 2011 *Urban Drainage*. 3rd edn. Spon Press, New York, USA.
- Cimorelli, L., Cozzolino, L., Covelli, C., Mucherino, C., Palumbo, A. & Pianese, D. 2013 Optimal design of rural drainage networks. *J. Irrig. Drain. Eng.* **139** (2), 137–144.

- Cimorelli, L., Cozzolino, L., Covelli, C., Della Morte, R. & Pianese, D. 2014 Enhancing the efficiency of the automatic design of rural drainage networks. *J. Irrig. Drain. Eng.* **140** (6), 04014015.
- Cimorelli, L., Morlando, F., Cozzolino, L., Covelli, C., Della Morte, R. & Pianese, D. 2015 Optimal positioning and sizing of detention tanks within urban drainage networks. *J. Irrig. Drain. Eng.* **142** (1), 04015028.
- Cisty, M. 2010 Hybrid genetic algorithm and linear programming method for least-cost design of water distribution systems. *Water Resour. Manage.* **24** (1), 1–24.
- Cozzolino, L., Cimorelli, L., Covelli, C., Mucherino, C. & Pianese, D. 2015 An innovative approach for drainage network sizing. *Water* **7** (2), 546–567.
- Demetrescu, C., Goldberg, A. & Johnson, D. 2006 9th DIMACS implementation challenge – shortest paths. Available at: [www.dis.uniroma1.it/challenge9/](http://www.dis.uniroma1.it/challenge9/).
- Elimam, A., Charalambous, C. & Ghobrial, F. 1989 Optimum design of large sewer networks. *J. Environ. Eng.* **115** (6), 1171–1190.
- Fu, G., Makropoulos, C. & Butler, D. 2010 Simulation of urban wastewater systems using artificial neural networks: embedding urban areas in integrated catchment modelling. *J. Hydroinform.* **12** (2), 140–149.
- Haghighi, A. & Bakhshipour, A. E. 2012 Optimization of sewer networks using an adaptive genetic algorithm. *Water Resour. Manage.* **26** (12), 3441–3456.
- Haghighi, A. & Bakhshipour, A. E. 2015 Deterministic integrated optimization model for sewage collection networks using Tabu search. *J. Water Resour. Plann. Manage.* **141** (1), 04014045.
- Haghighi, H., Samani, M. V. & Samani, Z. 2011 GA-ILP method for optimization of water distribution networks. *Water Resour. Manage.* **25** (7), 1791–1808.
- Haith, A. 1966 Vertical Alignment of Sewer and Drainage Systems by Dynamic Programming. Master thesis, Massachusetts Institute of Technology, Boston, MA, USA.
- Holland, M. 1966 Computer Models of Wastewater Collection Systems. Master thesis, Harvard University, Cambridge, MA, USA.
- Kulkarni, V. S. & Khanna, P. 1985 Pumped wastewater collection systems optimization. *J. Environ. Eng.* **111** (5), 589–601.
- Li, G. & Matthew, R. 1990 New approach for optimization of urban drainage systems. *J. Environ. Eng.* **116** (5), 927–944.
- Marchionni, V., Lopez, N. & Mamouros, L. 2014 Modelling sewer systems costs with multiple linear regression. *Water Resour. Manage.* **28** (13), 4415–4431.
- Moieni, R. & Afshar, M. H. 2012 Layout and size optimization of sanitary sewer network using intelligent ants. *Adv. Eng. Software* **51**, 49–62.
- Mortazavi-Naeini, M., Kuczera, G. & Cui, L. 2015 Efficient multi-objective optimization methods for computationally intensive urban water resources models. *J. Hydroinform.* **17** (1), 36–55.
- Navarro, I. 2009 Optimized Network of Urban Drainage Design. Master thesis, Universidad de los Andes, Bogotá, Colombia.
- Palumbo, A., Cimorelli, L., Covelli, C., Cozzolino, L., Mucherino, C. & Pianese, D. 2014 Optimal design of urban drainage networks. *Civil Eng. Environ. Syst.* **31** (1), 79–96.
- Pan, T. & Kao, J. 2009 GA-QP model to optimize sewer system design. *J. Environ. Eng.* **135** (1), 17–24.
- Sanchez, A., Medina, N., Vojinovic, Z. & Price, R. 2014 An integrated cellular automata evolutionary-based approach for evaluating future scenarios and the expansion of urban drainage networks. *J. Hydroinform.* **16** (2), 319–340.
- Sousa, J., Ribeiro, A., Cunha, M. C. & Antunes, A. 2002 An optimization approach to wastewater systems planning at regional level. *J. Hydroinform.* **4** (2), 115–123.
- Swamee, P. K. & Sharma, A. K. 2013 Optimal design of a sewer line using linear programming. *Appl. Math. Modell.* **37** (6), 4430–4439.
- Yeh, S. F., Chu, C. W., Chan, Y. J. & Lin, M. D. 2011 Applying Tabu search and simulated annealing to the optimal design of sewer networks. *Eng. Optim.* **43** (2), 159–174.

First received 15 May 2015; accepted in revised form 3 March 2016. Available online 2 April 2016