water

MDPI

Article

An Iterated Local Search Algorithm for Multi-Period Water Distribution Network Design Optimization

Annelies De Corte * and Kenneth Sörensen

ANT/OR Operations Research Group, University of Antwerp, Prinsstraat 13, 2000 Antwerp, Belgium; kenneth.sorensen@uantwerpen.be

* Correspondence: annelies.decorte@uantwerpen.be; Tel.: +32-3-265-4040

Academic Editors: Zoran Vojinovic and Michael B. Abbott

Received: 2 June 2016; Accepted: 10 August 2016; Published: 20 August 2016

Abstract: Water distribution networks consist of different components, such as reservoirs and pipes, and exist to provide users (households, agriculture, industry) with high-quality water at adequate pressure and flow. Water distribution network design optimization aims to find optimal diameters for every pipe, chosen from a limited set of commercially available diameters. This combinatorial optimization problem has received a lot of attention over the past forty years. In this paper, the well-studied single-period problem is extended to a multi-period setting in which time varying demand patterns occur. Moreover, an additional constraint—which sets a maximum water velocity—is imposed. A metaheuristic technique called iterated local search is applied to tackle this challenging optimization problem. A full-factorial experiment is conducted to validate the added value of the algorithm components and to configure optimal parameter settings. The algorithm is tested on a broad range of 150 different (freely available) test networks.

Keywords: water distribution network design; iterated local search; metaheuristic; mixed-integer non-linear programming; pipe sizing

1. Introduction

1.1. Access to Drinking Water

A safe, adequate, and accessible supply of drinking water is one of the basic necessities of any human being. According to a study from the World Health Organization (WHO), improving access to safe potable water not only reduces the overall risk of disease, but can also be an effective part of poverty reduction strategies [1].

The most efficient and effective way to transport drinking water is through a network of pipelines. The first water distribution networks date back to the ancient Greeks, the first civilization to use underground pipes for water supply [2]. In 2010, water distribution networks seem omnipresent, but nonetheless, 11% of the world's population does not have access to an improved drinking water source (i.e., a drinking water source that is protected from outside contamination), and only 54% of the world's population has piped drinking water on premises [3].

A large majority of all water distribution networks are maintained by water distribution companies. These operate in a dynamic, evolving setting as a result of three main evolutions: **changes in water usage:** population growth in certain parts of the world, combined with increasing living standards (houses equipped with sanitary installations, washing and dish washing machines, etc.) leads to a rising water consumption. A trend in the opposite direction is the growing consciousness of the scarcity of natural resources, which leads to a more rational use of water. Intermittent water supply also poses a challenge, since this is not only inconvenient for end-users, but also affects water quality and leads to a higher probability of pipe breaks and leakages. **Development of dual systems:**

around 80% of the water supplied to residences is used for activities other than human consumption, such as sanitary services and landscape irrigation. This urges source separation and dual systems that convey water of different levels of quality according to their end use: high quality water is delivered in smaller pipes for consumption, whereas the so-called "grey-water systems" transport recycled water for lower quality needs. The use of high-quality drinking water as firewater will decrease, which reduces the need for over-sized water distribution pipes [4]. **Urban development and renewal:** urban growth, increased attention for livability, and sustainable cities are some of the key drivers for urban (redevelopment), which also includes the (re)design of water distribution networks.

1.2. Decision Support Tools

The trends mentioned in the previous paragraph force water distribution companies to rethink their network configuration by modifying or expanding their current distribution systems. Construction or reorganization of these networks requires major investments. Hence, an efficient layout, design, and planning of water distribution networks is of crucial importance. Decision support systems are therefore potentially very useful in supporting water distribution companies in these decision making processes. This paper aims to develop a decision support system that supports network design decisions. These are decisions related to the optimal type of pipe connecting the supply, demand, and junction nodes in the distribution network. Such decisions are made when pipes need to be (re)placed or rehabilitated due to aging, changing demographic settings, network redesign, or network expansion.

Traditionally, such design decisions are made based on expert experience. When networks increase in size, however, it is doubtful that the applied rules of thumb will lead to optimal decisions. Hence, the use of decision support systems (DSS) could result in significant savings. Decision making in this area could benefit from the use of such a DSS in two ways. First, it may improve the *efficiency* with which a user makes a decision, since alternative feasible solutions are generated faster than using a manual process. Second, the *effectiveness* of the decision process may also be improved, as for problems with a huge solution space, algorithms generally find better solutions than human brainpower.

1.3. Water Distribution Network Design Optimization

As stated in the previous paragraph, the focus of this paper is finding optimal network designs. Figure 1 illustrates the water distribution network design optimization problem. A toy network that contains 60 demand nodes, 1 water supply reservoir and 83 water distribution pipes has to be designed. In the network design phase, network topology is assumed given, as can be seen in Figure 1a. The aim is to find the best suited diameter for every pipe in this network. Assume that there are 16 possible pipe diameters, ranging from 20 mm up to 1000 mm. Since all possible diameters can be selected for every pipe, 16^{83} possible configurations (or combinations of diameters) could be designed. Finding the lowest-cost design (or the optimal diameter size for every pipe) while respecting hydraulic laws and customer requirements is not straightforward—even for such a small network. The right side of the figure illustrates the optimized network, where diameters are assigned to every pipe. Thicker lines represent larger diameters in the figure.

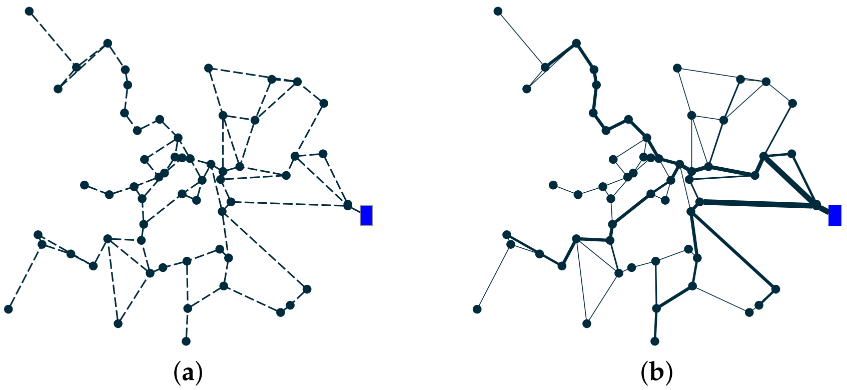


Figure 1. Example of the water distribution network design optimization problem for a small network. (**a**) Unoptimized network; (**b**) Optimized network. Thicker lines represent larger diameters.

1.4. State of the Art

Water distribution network design optimization has been an active research field for over four decades. A large majority of papers in this area focus on the single-period, single-objective, gravity-fed design optimization problem. Early work applies more traditional mathematical programming methods, such as *linear programming (LP)*, originally applied by Alperovits and Shamir [5] and extended by Quindry et al. [6] and Kessler and Shamir [7]; or *non-linear programming (NLP)* in Shamir [8], El-Bahrawy and Smith [9], Fujiwara and Khang [10], and Duan et al. [11].

In the last three decades, considerable attention has been given to the development of (meta)heuristic algorithms to solve the water distribution network design optimization problem. These techniques use a hydraulic solver (usually EPANET 2.0 [12]) to solve the hydraulic equations externally, while the heuristic manipulates the selected pipe types. *Local search metaheuristics* operate on a single solution and try to improve this solution in small steps. Loganathan [13] and Cunha and Sousa [14] applied simulated annealing, and Cunha et al. [15] applied a tabu search. *Constructive metaheuristics* iteratively construct solutions, rather than improving complete solutions. Ant colony optimization, for example, is applied by Maeier et al. [16], and ant systems by Zecchin et al. [17]. However, *population-based metaheuristics* are the most popular in this field. These algorithms operate on a population of solutions and combine them into new solutions. Among this class of metaheuristics are genetic algorithms in Dandy et al. [18], Gupta et al. [19], and Bi et al. [20], scatter search in Lin et al. [21], and differential evolution in Vasan and Simonovic [22]. A more recent paper by Marchi et al. [23] does not develop a new method, but introduces a methodology to compare various evolutionary algorithms for water distribution network design. The authors show that algorithm performance depends on the problem characteristics and the number of function evaluations. Moreover, the importance of correct calibration is stressed. A more detailed overview of the state-of-the-art in this research area is given in De Corte and Sörensen [24].

Some research has already been dedicated to the extensions formulated in this paper. Farmani et al. [25], for example, also perform a *multi-period optimization*, but formulate the design problem as a multi-objective optimization problem. They define maximizing network reliability as a second objective next to cost minimization, and apply a multi-objective evolutionary algorithm to optimize the Anytown network. Gupta et al. [19] and Bragalli et al. [26] also apply a *velocity constraint* on the water flowing through the distribution pipes. The first apply a genetic algorithm on six (small) instances while the latter use mathematical programming on bigger, closer-to-reality instances.

1.5. Contributions

The contribution of this paper is twofold. A first contribution is that the well-studied single-period, pressure-constrained model is extended, and therefore, closer to reality. This work adjusts this model in two ways. The single-period setting is extended to a multi-period set-up, in which every demand node has a 24 h water demand pattern. Moreover, an additional constraint—which imposes a limit on the maximal velocity of water through the pipes—is introduced. These additions are formulated in Section 2.

A second contribution is related to algorithm composition, configuration, and testing. The developed algorithm is developed so that only components that show a significant added value are included in its final configuration. A clear and systematic test procedure is used to determine these components. Moreover, the algorithm contains some parameters that need to be configured. A full-factorial experiment is performed to determine the optimal level for each parameter. This allows solid conclusions to be drawn regarding the effects of algorithm configuration and parameter settings on solution quality and computing time. More details are given in Section 3.3. In terms of algorithm testing, the algorithm is tested on a broad set of challenging HydroGen test networks. HydroGen (De Corte and Sörensen [27]) is a tool to automatically generate water distribution test networks of varying size and characteristics. By using an extensive set of test networks, we are able to draw robust conclusions on the performance of the algorithm. Moreover, these HydroGen networks are free to use and available online, which should foster easy comparison of results. The experimental results are given in Section 4.

2. Model Formulation

A water supply system consists of a transmission and a distribution system. The focus in this paper is on the distribution network. Moreover, the considered networks are all gravity-fed, which means that the water is supplied by gravitational forces and no pumps are needed. Water quality considerations are not taken into account. This is in line with previous research in this area, where a majority of the work focuses on the well-defined single-period, single-objective, gravity-fed water distribution network optimization problem. As stated above, This research extends this problem in two ways: (1) by imposing an additional constraint (i.e., a maximal velocity of the water in the distribution pipes); and (2) by applying time-varying diurnal demand patterns. This problem will therefore be formulated as the single-objective, multi-period, gravity-fed water distribution network design (WDND) optimization problem with pressure and velocity constraints.

2.1. Mathematical Formulation

To formulate the water distribution network design problem as a mathematical model, the water distribution network is represented as a connected graph with a set of water demand or supply nodes $N = \{n_1, n_2, \ldots\}$ and a set of water distribution pipes $P = \{p_1, p_2, \ldots\}$. The set of closed loops in this graph is denoted $L = \{l_1, l_2, \ldots\}$. The objective of this WDND optimization problem is to minimize the total investment cost (TIC) of the network design by selecting the optimal pipes from a set of available pipe types. The cost of an individual pipe depends on the type t that is chosen for this pipe from the list of commercially available types $T = \{t_1, t_2, \ldots\}$. A pipe's type determines both its diameter and the material from which it is made, which in turn determine its hydraulic properties. If the cost per meter of a pipe p of type t is represented by IC_t, and the length of pipe p is represented as L_p, the objective function of the multi-period, gravity-fed water distribution network design optimization problem can be written as:

$$\min TIC = \sum_{p \in P} \sum_{t \in T} L_p \; IC_t \; x_{p,t} \tag{1}$$

where $x_{p,t}$ is the binary decision variable that determines whether pipe p is of type t ($x_{p,t} = 1$) or not ($x_{p,t} = 0$). Since the network layout is assumed to be given, L_p is known. The investment cost IC_t is also given for every commercially available pipe type. This objective function expresses the minimization of the total network cost.

The objective function is limited by physical mass and energy conservation laws, by minimum head requirements in the demand nodes, and by maximal water velocities in the pipes, for every time period $\tau \in \mathcal{T}$.

The *mass conservation law* must be satisfied for each node $n \in N = \{n_1, n_2, \ldots\}$ in every time period $\tau \in \mathcal{T}$. This law states that the volume of water flowing into a node in the network per unit of time must be equal to the volume of water flowing out of this node. Let $Q_{(n_1,n),\tau}$ represent the amount of water flowing from node n_1 to node n in time τ, and let $WS_{n,\tau}$ be the external water supply and $WD_{n,\tau}$ the external water demand of node n in period τ (all expressed in m^3/s); then, the following should hold:

$$\sum_{n_1 \in N/n} Q_{(n_1,n),\tau} - \sum_{n_2 \in N/n} Q_{(n,n_2),\tau} = WD_{n,\tau} - WS_{n,\tau} \qquad \forall n \in N \qquad \forall \tau \in \mathcal{T} \tag{2}$$

Furthermore, for each closed loop $l \in L = \{l_1, l_2, \ldots\}$, the *energy conservation law* must be satisfied for every time period τ. This law states that the sum of pressure drops in a closed loop is zero. Pressure drops (or head losses) in piping systems are mainly caused by wall shear in pipes. The energy conservation law can be stated as:

$$\sum_{p \in l} \Delta H_{p,\tau} = 0 \qquad \forall l \in L \qquad \forall \tau \in \mathcal{T} \tag{3}$$

with $\Delta H_{p,\tau}$ representing the head loss in pipe p in time τ. Head losses in the pipes of the network are approximated using Hazen–Williams equations, with the parameters set to the values used by EPANET 2.0, the hydraulic solver used in this paper:

$$\Delta H_{p,\tau} = \frac{10.6668\; y_{p,\tau}\; Q_{p,\tau}^{1.852}\; L_p}{\sum\limits_{t \in T} (x_{p,t}\; C_t^{1.852}\; D_t^{4.871})} \tag{4}$$

Consequently, Equation (3) can be rewritten as:

$$\sum_{p \in l} \left[\frac{10.6668\; y_{p,\tau}\; Q_{p,\tau}^{1.852}\; L_p}{\sum\limits_{t \in T} (x_{p,t}\; C_t^{1.852}\; D_t^{4.871})} \right] = 0 \qquad \forall l \in L \qquad \forall \tau \in \mathcal{T} \tag{5}$$

In this equation, $y_{p,\tau}$ is the sign of $Q_{p,\tau}$, which is the amount of water flowing through pipe p (in m^3/s) in time τ. This sign incorporates changes in the water flow direction relative to the defined flow directions. L_p is the pipe length (in m), C_t is the Hazen–Williams roughness coefficient of pipe type t (unit-less), and D_t is the diameter of pipe type t (in m). Parameters D_t and C_t are determined by the type of pipe and are assumed given for each available type. Note that $y_{p,\tau}$ and $Q_{p,\tau}$ represent alternative formulations of $Q_{(n_1,n_2),\tau}$ if pipe p connects n_1 and n_2.

Minimum pressure head requirements exist for every (demand) node $n \in N$ at every time period $\tau \in \mathcal{T}$. Let $H_{n,\tau}$ be the pressure head in node n (in m) at time τ, and $H_{n,\tau}^{min}$ the minimum pressure head in node n (in m) for time τ. This constraint can be represented as:

$$H_{n,\tau}^{min} \leq H_{n,\tau} \qquad \forall n \in N \qquad \forall \tau \in \mathcal{T} \tag{6}$$

A *maximal water velocity* limit is present for every pipe $p \in P$. The velocity of the water flowing through pipe p in time period τ, $v_{p,\tau}$ cannot exceed the imposed maximal velocity $v_{p,\tau}^{max}$:

$$v_{p,\tau} \leq v_{p,\tau}^{max} \qquad \forall p \in P \qquad \forall \tau \in \mathcal{T} \tag{7}$$

with:

$$v_{p,\tau} = \frac{1.27\ Q_{p,\tau}}{\sum\limits_{t \in T} (x_{p,t}\ D_t^2)} \qquad \forall \tau \in \mathcal{T} \tag{8}$$

$Q_{p,\tau}$ is the amount of water flowing through pipe p in time τ, $x_{p,t}$ the binary decision variable, and D_t the diameter of pipe type t, which is given for each available pipe type.

Moreover, every pipe p should get exactly one type t assigned:

$$\sum_{t \in T} x_{p,t} = 1 \qquad \forall p \in P \tag{9}$$

From this mathematical formulation, it is clear that the WDND optimization problem is a mixed-integer, non-linear optimization problem, which puts this problem out of reach of exact linear or mixed-integer programming solvers like CPLEX or Gurobi. Moreover, this problem is a *combinatorial* optimization problem, since its decision variables (i.e., the type of each pipe) are discrete. The problem has also been proven to be NP-hard (nondeterministic polynomial time hard) by Yates [28].

2.2. Coping with the Extensions

2.2.1. Extension to Multi-Period Setting

The multi-period extension adds an extra layer of complexity to the WDND optimization problem, which is evident by the fact that all the flow-related variables receive an extra index for time period τ. An easy and intuitive solution to cope with this added complexity is to reduce the dynamic, multi-period setting to a static, single-period setting. Two different approaches to do so—*maxPeriod* and *maxDemand*—are compared below, and it is shown that such reductions are not possible.

maxPeriod

A first simplification strategy is to run a single-period model for the time period in which the total network demand is at its highest. This approach to reducing the multi-period problem to a single-period setting fails. The reason is that some areas will generally have a low water demand during the calculated maximal period, which will lead to small diameters for this area when optimizing the network using the single-period model. When this optimized network design is simulated in the multi-period setting, pressure constraints will be violated in these areas. When water demand increases in this area in other time periods, the small diameters will lead to high pressure drops, and consequently, a violation of the minimal pressure head constraint. This is illustrated in Figure 2. In (a), the network is optimized with the multi-period algorithm described in Section 3. During the multi-period simulation, no pressure deficits occur. In (b), the network is optimized according to the *maxPeriod* strategy. This will lead to a lower cost network, but during the multi-period simulation, some nodes will experience a pressure deficit. These nodes are visualized by the larger (red colored) demand nodes in (b).

maxDemand

An alternative approach would be to reduce the dynamic model to a single-period model in which water demand is maximal. For every node, the highest demand that is encountered during the multi-period timeframe is set as the actual demand in the static model. This approach will only be successful when there is only one demand pattern, which is far from realistic. Unavoidably, as soon as there is variation in the demand patterns, the designs based on this single-period setting will lead to over-sized (and therefore, more expensive) designs. As can be seen in Figure 2c, the design—based on the static *maxDemand* model—does not lead to any pressure violation in the dynamic setting. The design, however, is considerably more expensive than the design found by the multi-period model.

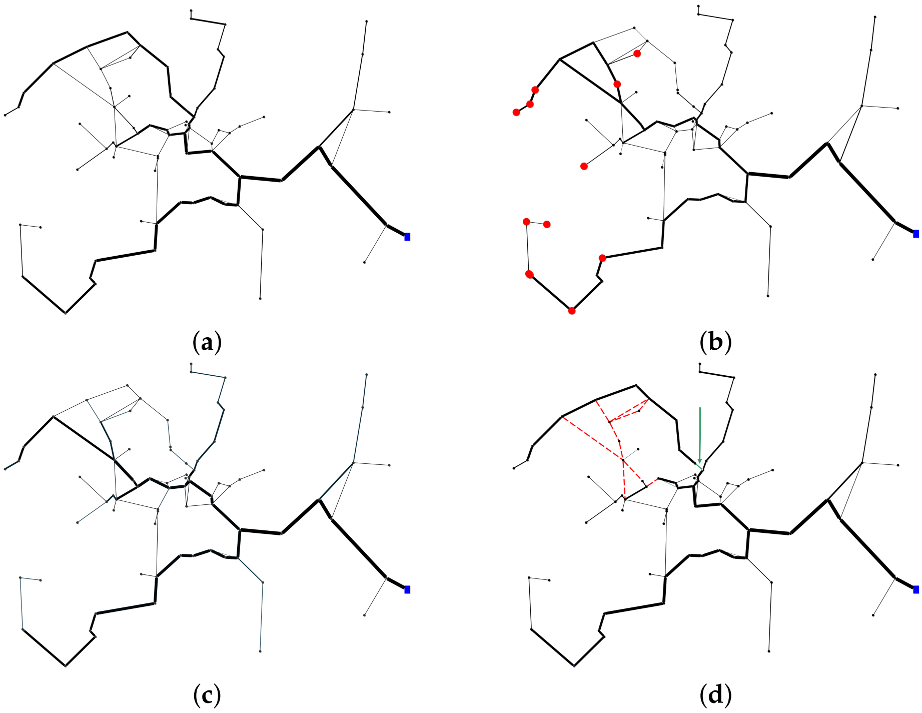


Figure 2. Network optimized with (**a**) multi-period simulation (total investment cost, TIC = 1.232 × 10^6 *EUR*); (**b**) single-period optimization using *maxPeriod* (TIC = 0.967 × 10^6 *EUR*); (**c**) single-period optimization using *maxDemand* (TIC = 1.270 × 10^6 *EUR*); Network (**d**) shows that changing a single pipe causes velocity violations in other (non-neighboring) pipes.

2.2.2. Constraint on Velocity in Pipes

The water velocity in a pipe is a function of (among other factors) the diameter of the pipe. Consequently, altering the diameter of a pipe leads to a change of the water velocity in the pipe. Hence, when a pipe diameter is altered, a velocity check has to be performed. (Un)fortunately, decreasing a single pipe's diameter affects the entire network flow, and therefore, the velocity needs to be checked in every pipe of the network, not just the one whose diameter was changed. This is visible in Figure 2d. If the indicated (blue) pipe is decreased in diameter, velocity violations will occur in the pipes colored in red.

3. An Iterated Local Search Algorithm

In this section, the WDND optimization problem is tackled with a metaheuristic technique, called iterated local search. Iterated local search (ILS) is a straightforward, easy to implement, and easy to adjust heuristic optimization algorithm. These characteristics make it especially suitable as the underlying optimization engine in a decision support system, since decision makers are understandably less reluctant to use more transparent decision support tools. Despite its simplicity, ILS is broadly applied in different research fields and as the basis of many state-of-the-art algorithms for problems such as travelling salesperson problems, scheduling problems, graph partitioning problems, etc. [29].

ILS is a *local search* metaheuristic: it operates on a single solution and tries to improve this solution in small steps during the local search. Figure 3, for example, shows different candidate solutions or network configurations that could be created during the local search. These solutions are networks for which diameters are assigned to every pipe. These solutions can also be represented by a vector, where every element of the vector corresponds to the diameter that is selected for that pipe, as can be seen in Table 1. This table gives vector representations of the candidate solutions in Figure 3. Every solution corresponds to a total investment cost (TIC)—calculated via the objective function, Equation (1), and using the input data from Figure 4—and is either hydraulically feasible (satisfying all constraints mentioned in Section 2) or infeasible (violating one or more of the constraints). Candidate solution 2, for example, violates the minimum pressure constraint in node 7, and candidate solution 4 violates the maximum velocity constraint in pipes 2 and 4. Therefore, both solutions are labelled infeasible. ILS is an *iterated* local search, implying that it alternates the local search phase with a perturbation phase. During the local search phase, small improvements (or changes in diameters that lead to lower cost, feasible solutions) are made to the current solution (which is a potential network design) until a local optimum is reached, meaning that there is no more improvement possible. Next, a perturbation is performed. This perturbation temporarily makes the solution worse by randomly changing some pipe diameters, with the aim of leading to better solutions in the consecutive local search phase. A more detailed algorithm description is given in Section 3.2.

ILS can therefore be understood as a combination of intensification or exploitation by the use of the local search, and diversification or exploration by the perturbation.

Table 1. Different solutions or network designs in vector representation.

Candidate	**Pipe ID**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Feasibility**
Solution	**Length (in m)**	**31**	**20**	**35**	**37**	**24**	**50**	**12**	**65**	**TIC**
candidate solution 1	diameter (in mm) cost (in *EUR*)	150 1550	150 1000	80 1225	80 1295	100 912	60 1100	60 264	80 2275	feasible 9621 *EUR*
candidate solution 2	diameter (in mm) cost (in *EUR*)	150 1550	150 1000	80 1225	60 814	100 912	60 1100	60 264	80 2275	infeasible 9140 *EUR*
candidate solution 3	diameter (in mm) cost (in *EUR*)	150 1550	150 1000	60 770	80 1295	100 912	60 1100	60 264	80 2275	feasible 9166 *EUR*
candidate solution 4	diameter (in mm) cost (in *EUR*)	150 1550	100 760	80 770	80 1295	100 912	60 1100	60 264	80 2275	infeasible 8926 *EUR*

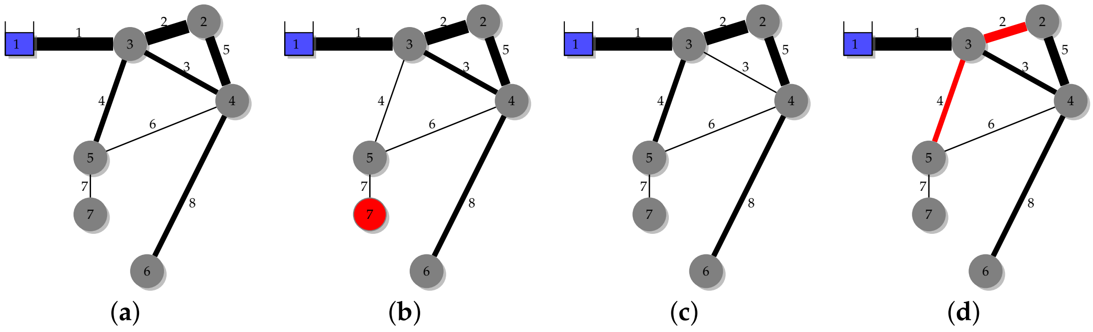
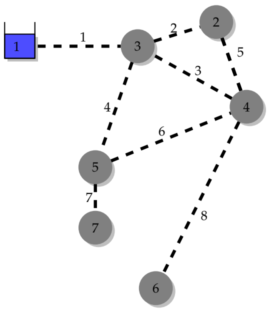


Figure 3. Different solutions or network designs in network representation. (**a**) Candidate solution 1; (**b**) Candidate solution 2; (**c**) Candidate solution 3; (**d**) Candidate solution 4.



Pipe ID	**Length in m**
1	31
2	20
3	35
4	37
5	24
6	50
7	12
8	65

Diameter **in mm**	**Roughness** ***unitless***	**Cost** **in *EUR* per m**
60	130	22
80	130	35
100	130	38
150	130	50
200	130	61

Figure 4. Problem instance: unoptimized network topology (**left**); pipe lengths (**center**); and available pipe types with their corresponding costs (**right**).

3.1. Implementation

To enable a straightforward comparison with existing methods, all hydraulic equations are solved by EPANET 2.0, which is the hydraulic solver applied in most existing works. The ILS algorithm is implemented in C++. The interaction between the ILS algorithm and the EPANET simulation engine is established by the use of an extended EPANET toolkit, developed by M. López-Ibáñez [30,31]. As visualized in Figure 5, two different input files are needed. The first gives specifications on the available pipe types (diameter, cost, roughness coefficient) and specifies minimal head requirements for the demand nodes and maximal water velocity constraints for the pipes. The second file describes the network in EPANET input format, and contains the network topology and the characteristics of pipes and nodes. These two files serve as input for the ILS algorithm and the hydraulic solver.

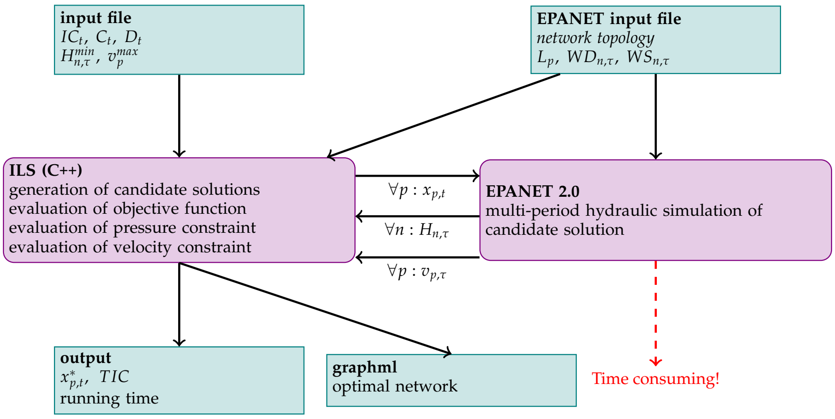


Figure 5. Interaction between iterated local search (ILS) and EPANET 2.0.

Every time a candidate solution is generated by the algorithm, this information (which basically is a candidate network design) is sent to EPANET 2.0 and hydraulically simulated for every time period. This simulation yields hydraulic results, which are used by the algorithm to evaluate the problem constraints.

The final output will be an optimized network ($x^*_{p,t}$) with a corresponding cost TIC. This network is also outputted in GraphMl format, to enable easy visualization, since this benefits the process of solution delivery.

3.2. Detailed Algorithm Description

The developed ILS algorithm consists of six steps, which are explained in more detail below. Figure 6 depicts the corresponding flow chart.

Sort

In a preliminary step, the set of pipes is sorted. This sorting of the pipes determines the order in which the local search adjusts the pipe diameters and how the algorithm moves through the solution space. Two different sorts are tested. The first, `lengthSort`, is similar to the sorting that was applied in the single-period algorithm and sorts the set of pipes according to decreasing length. The second, `costSort`, sorts the pipes according to decreasing cost savings that could be made by decreasing the diameter of that pipe by one size. When the pipes are sorted according to this decreasing cost difference, re-sorting of the pipe list is necessary during the local search.

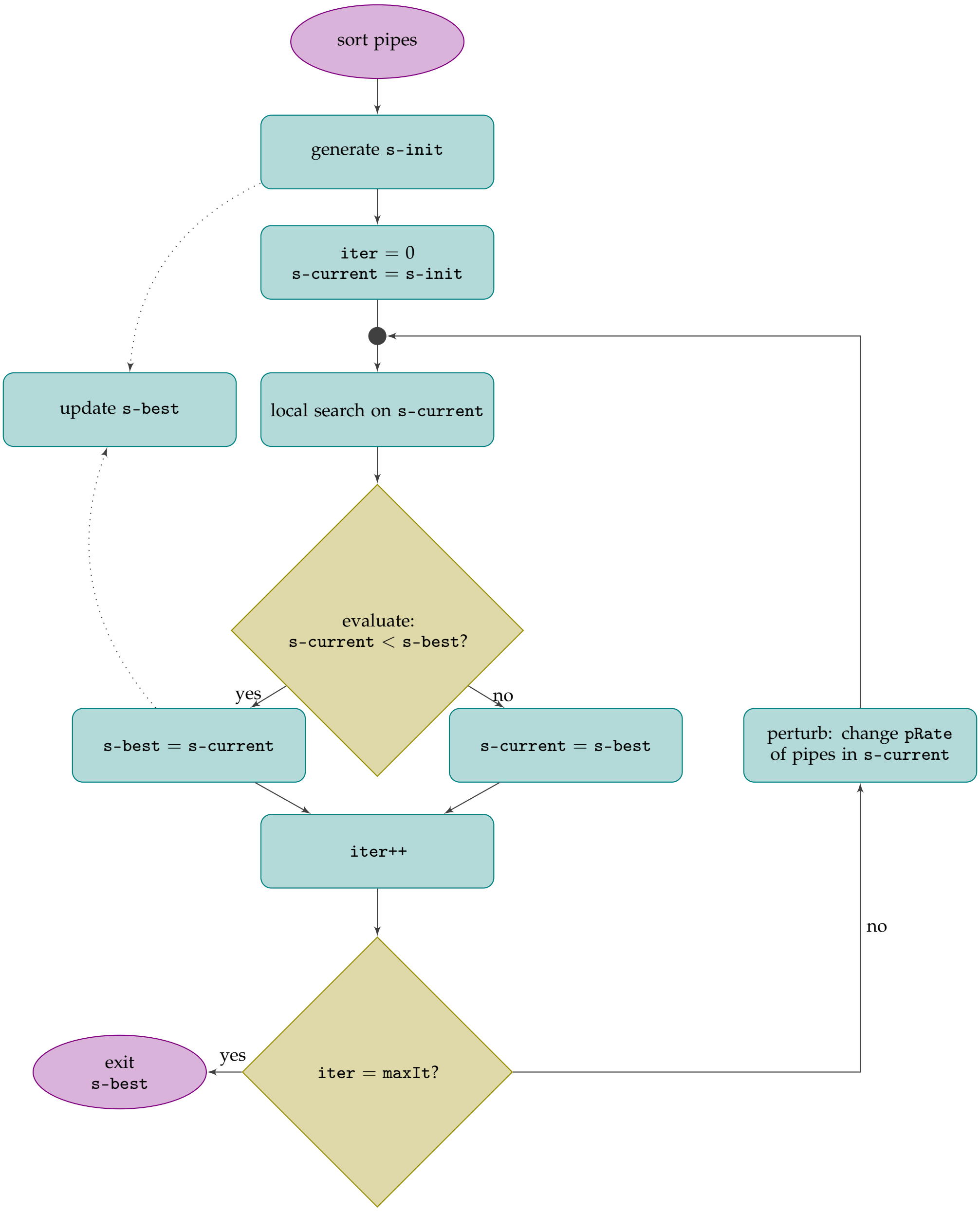


Figure 6. Flowchart of the iterated local search algorithm. s-best: initial global best solution; s-current: current solution; s-init: initial feasible solution.

Initial Solution

In a next step, an initial feasible solution, `s-init`, is constructed. A solution is considered feasible if all hydraulic constraints are strictly satisfied. Two different mechanisms are tested. The first, `highCost`, starts with a solution where all diameters are set at the largest available diameter. If this combination of diameters generates a solution that is feasible in all time periods, all pipe diameters are decreased by one size. This procedure is repeated until an infeasible solution is encountered: diameters are increased again and this results in an initial solution in which all pipes have the lowest diameter that is possible without violating the imposed constraints. The second mechanism, `lowCost`, is expected to be slower, but generates initial solutions with a lower cost than those generated by method `highCost`. First, all pipes are set to the smallest possible diameter. This solution is almost always infeasible (if it is feasible, it is optimal). Therefore, a second step consists of increasing the pipe diameters by going through the set of pipes and trying to increase each pipe's diameter by one size at a time and checking the hydraulic feasibility of this potential solution for every time period. Increasing the pipe diameter lowers (ceteris paribus) the head loss in that pipe, which could result in higher pressure heads at the nearby demand nodes. If this increase in diameter actually lowers the hydraulic deficit and does not violate the velocity constraint, it is applied. When there is no more deficit or velocity violation for the first time period, the procedure is repeated for the next time period. This procedure ends when a solution that is feasible in every time period is attained. This solution is set as the initial global best solution `s-best`. In the most extreme scenario, this solution will be one where all diameters are set to their maximal values.

Local Search

Next, a loop is performed until a stopping criterion is reached. The local search procedure is able to quickly improve the quality of the initially generated solutions by iteratively performing small changes—called *moves*—on the current solution. The *move type* that is applied in this local search (*decrease*) reduces the diameter of one pipe with one size at a time. Therefore, neighboring solutions have configurations in which all pipes but one have the same diameter as the current solution. Table 1 and Figure 3 visualize different (neighboring) solutions. Candidate solution 1 is visualized in Figure 1 and numerically visualized in Table 1. This solution can also be represented as a vector of pipe diameters: $\{150, 150, 80, 80, 100, 60, 60, 80\}$. Hydraulic simulation shows that candidate solution 1 is hydraulically feasible. A decrease move is applied on pipe 3 of candidate solution 1, implying that the diameter of that pipe will be decreased from 80 to 60 mm. This new candidate solution 2 is a neighbour of candidate solution 1; its vector representation is: $\{150, 150, 80, 80, 100, 60, 60, 80\}$. Simulation in EPANET 2.0 shows that candidate solution 2 does not violate any of the constraints. Therefore, this move leads to a feasible solution of lower total investment cost (the TIC decreased from 9621 € to 9140 €, as can be derived from Table 1). If we start from candidate solution 1 again and apply the decrease move on pipe 4, this diameter will decrease from 80 mm to 60 mm, leading to candidate solution 3. Hydraulic simulation of candidate solution 3 shows that this solution is infeasible, since it would violate the minimal pressure constraint in demand node 7, visualized in Figure 3. Another move could be the decrease of pipe 2 to a diameter of 100 mm. Hydraulic simulation shows that this leads to a violation of the maximal velocity constraint in pipes 2 and 4; candidate solution 4 is therefore also labeled as infeasible.

Deciding which pipe will be decreased in diameter depends on the applied pipe selection method. Two alternative pipe selection methods are tested. The first method, `noGrasp`, goes through the list of pipes successively. The other method, `grasp`, uses a greedy randomized adaptive search procedure to select which pipe should be changed in that current move. The larger the grasp size, the more random the pipe selection. When, for example, for a network of 200 pipes, the grasp size is set at 20%, a pipe will be uniformly randomly selected out of the first 40 pipes in the list. Note that a grasp size of 0% is equal to the `noGrasp` selection procedure. A *first improving* strategy is applied, meaning that as soon as a better (lower value for the objective function) feasible solution is encountered, `s-current` is replaced by this solution. A simple memory structure (`memory`) is introduced to speed up the algorithm. This list keeps track of all pipes that were changed unsuccessfully (i.e., violating the hydraulic constraints). It is prohibited to change these pipes' diameters again during the current search. The memory list is erased after every local search.

In the multi-period setting, the hydraulic simulation is even more time consuming than in the single-period setting, since a simulation has to be performed for every time period. Therefore, results are evaluated after every time period for the velocity and the pressure constraints. As soon as infeasible situations are encountered, the simulation is stopped in order to avoid (superfluous) simulations for the next time periods.

Evaluation and Acceptance

After every local search, the current solution, `s-current`, which is the local optimum of that local search (or the lowest cost feasible solution that can be reached by applying the *decrease* move), is evaluated. If the global best solution, `s-best`, is of lower cost, the current solution is replaced by `s-best`. If the current solution is of lower cost than the global best solution, `s-best`, this global best solution is replaced by the current solution.

Perturbation

When a local optimum is reached, a perturbation move is used to jump out of this local optimum. The perturbation move is performed on the current solution, `s-current`, which is at that moment equal to the global best solution, `s-best`. The perturbation takes a certain percentage—the so-called perturbation rate—of randomly selected pipes, and increases the diameters of these pipes by one size. For every change in diameter, a hydraulic check is performed. When such a change violates the pressure and/or velocity constraints, it is reversed. This perturbed solution is therefore always hydraulically feasible and is the starting solution for the next local search. The optimal perturbation rate balances between two extremes: if too many pipes are changed, the procedure reduces to a random (feasible) restart—which is not necessarily bad. If too few pipes are changed, the algorithm is unable to escape from local optima.

Termination Criterion

The algorithm is stopped after a certain maximal number of iterations. The lowest cost solution encountered by the algorithm is `s-best`.

As stated above, it is important to validate whether all of the aforementioned mechanisms have an added value (meaning that they either lead to lower cost solutions, or else a decrease in algorithm running time). Some parameters can also take up different values or levels. Therefore, a full-factorial experiment is conducted to determine the best parameter settings. This experiment will lead to the optimal algorithm configuration, in terms of reaching low cost solutions and in terms of minimizing algorithm running time.

3.3. Optimal Algorithm Configuration

Two linear mixed-effects models are fitted to detect which factors or algorithm parameters significantly influence the algorithm's performance in terms of achieving low cost solutions and in terms of running time. Table 2 displays the analyzed factors and their corresponding levels. This analysis is performed on a set of 75 different test networks of different size (expressed by the number of pipes) and characteristics (reflected by the meshedness coefficient). Since the perturbation contains a random component, the algorithm will find different solutions when it is executed multiple times on the same problem instance. Therefore, the algorithm is executed five times on each test instance and for each factor combination. The average cost or time over these five runs is taken as the response variable. As a consequence, both data sets will contain 36,000 data points (75 instances × 480 parameter combinations). The first model quantifies the relationship between the achieved solution cost and the different parameters, the second model explains how much of the variance in the running time is explained by the analyzed parameters.

Table 2. Parameters and their tested values.

Parameter	**Levels**	**Value**
# Iterations	4	100, 200, 300, 400
# Pipes	5	100, 200, 300, 400, 500
Meshedness	3	0.10, 0.15, 0.20
Initial solution	2	`highCost`, `lowCost`
Pipe sort	2	`lengthSort`, `costSort`
Grasp size	3	0%, 10%, 20%
Memory list	2	`memory`, `noMemory`
Perturbation rate	5	1%, 10%, 20%, 30%, 40%

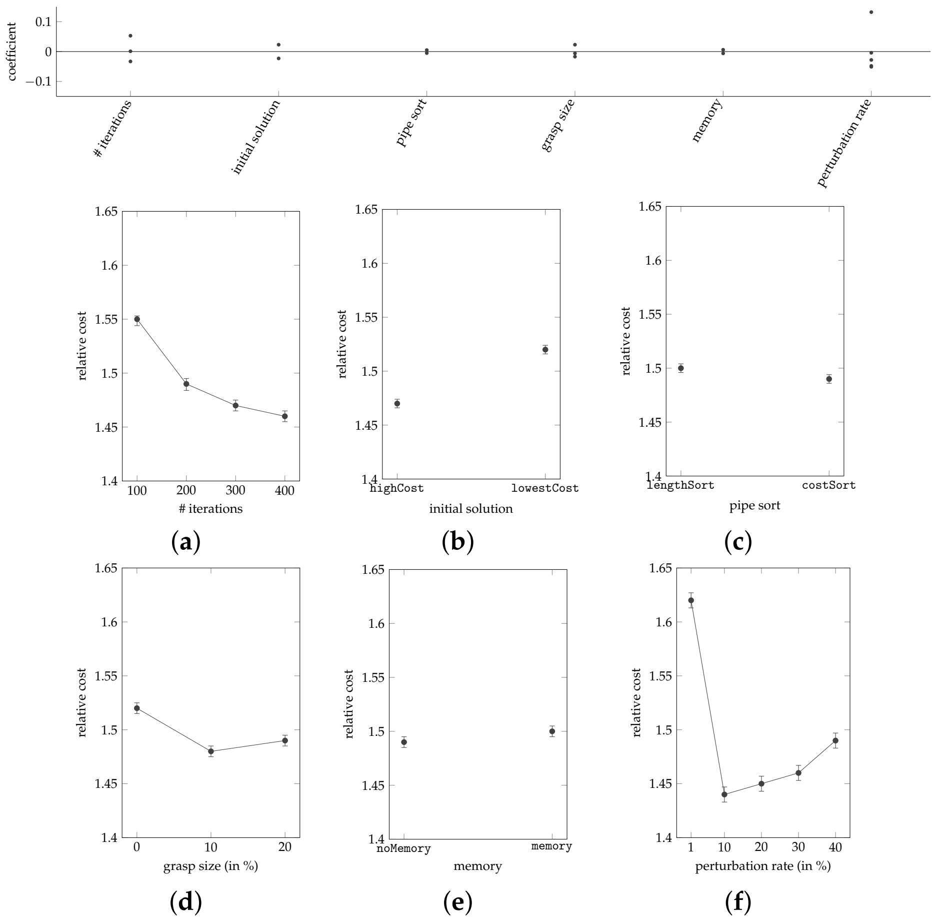
3.3.1. Model on the Relative Cost

The relative cost model with main effects has a coefficient of determination (*R*-squared adjusted) equal to 0.79, meaning that about 79% of the variation in the response variable (in this case, the relative cost) is explained by the independent variables. Adding interaction effects does not have a large effect on the adjusted *R*-squared— it is only increased to 0.81.

3.3.2. Model on the Running Time

For the estimation of the model on the running time, it seems reasonable to include the network size as a factor, since it can be expected that the size of the network will influence the running time. Therefore, the number of pipes was added as a factor in the model on the running time. The meshedness coefficient of the network is also included as a factor, to check whether the meshedness influences the algorithm running times—for example, if a higher amount of loops results in longer hydraulic simulations. The running time model with main effects only has a coefficient of determination (adjusted *R*-squared) of 0.77. It is clear that adding interaction effects significantly increases the explanatory power: with main and interaction effects included, the factors are able to explain 97% of the variance in the algorithm running time.

Figures 7 and 8 plot the values of the coefficients estimated for each factor in the main effects model on the cost and the running time, respectively. Figures 9 and 10 plot the values of the coefficients estimated for the factor interactions. The lower parts of these figures show the mean plots and confidence intervals for the means for all parameters. These mean plots visualize the exact effect of each of the significant parameters.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 7. Values of the coefficients and mean plots for the regression model on the *relative cost*: main effects.

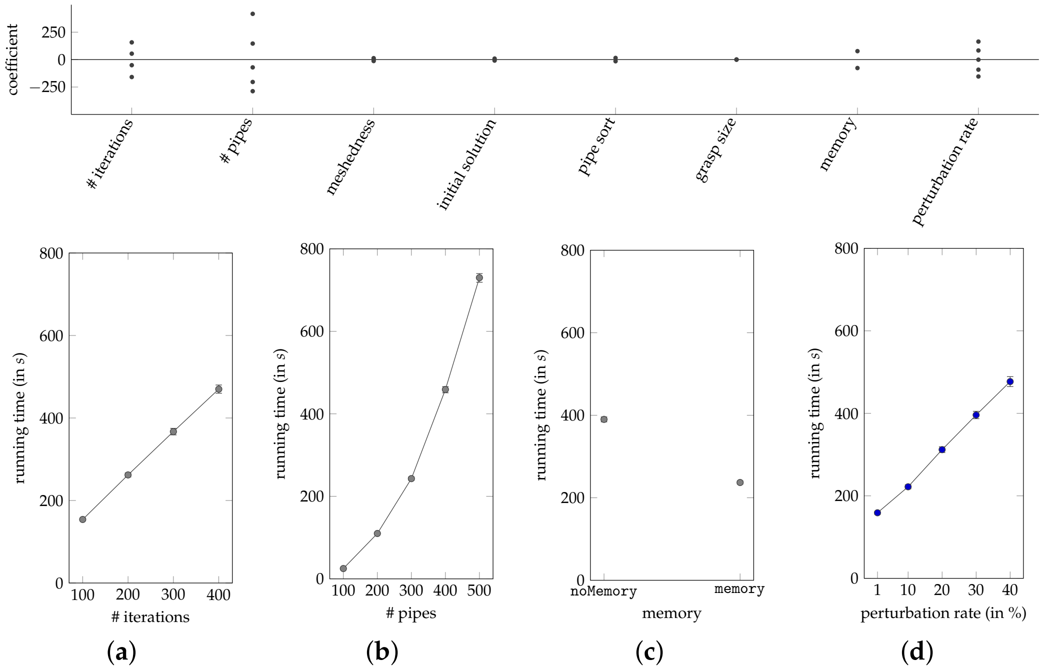


Figure 8. Values of the coefficients and mean plots for the regression model on the *running time*: main effects.

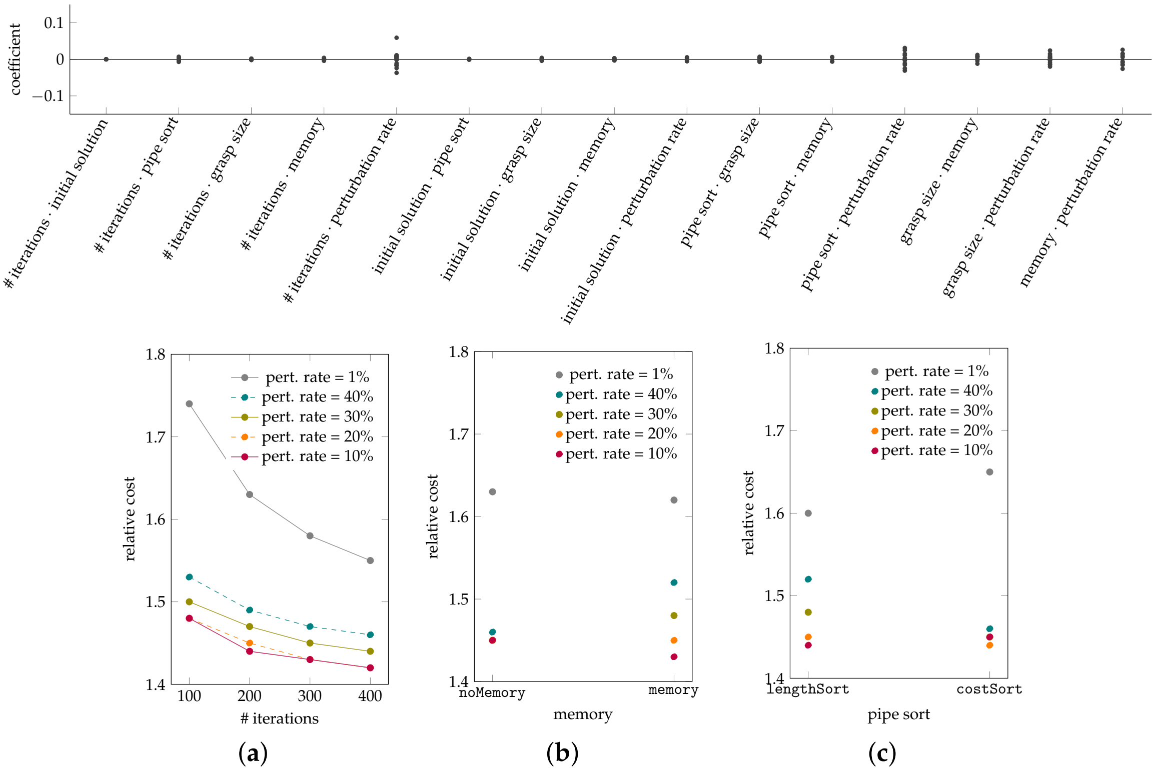
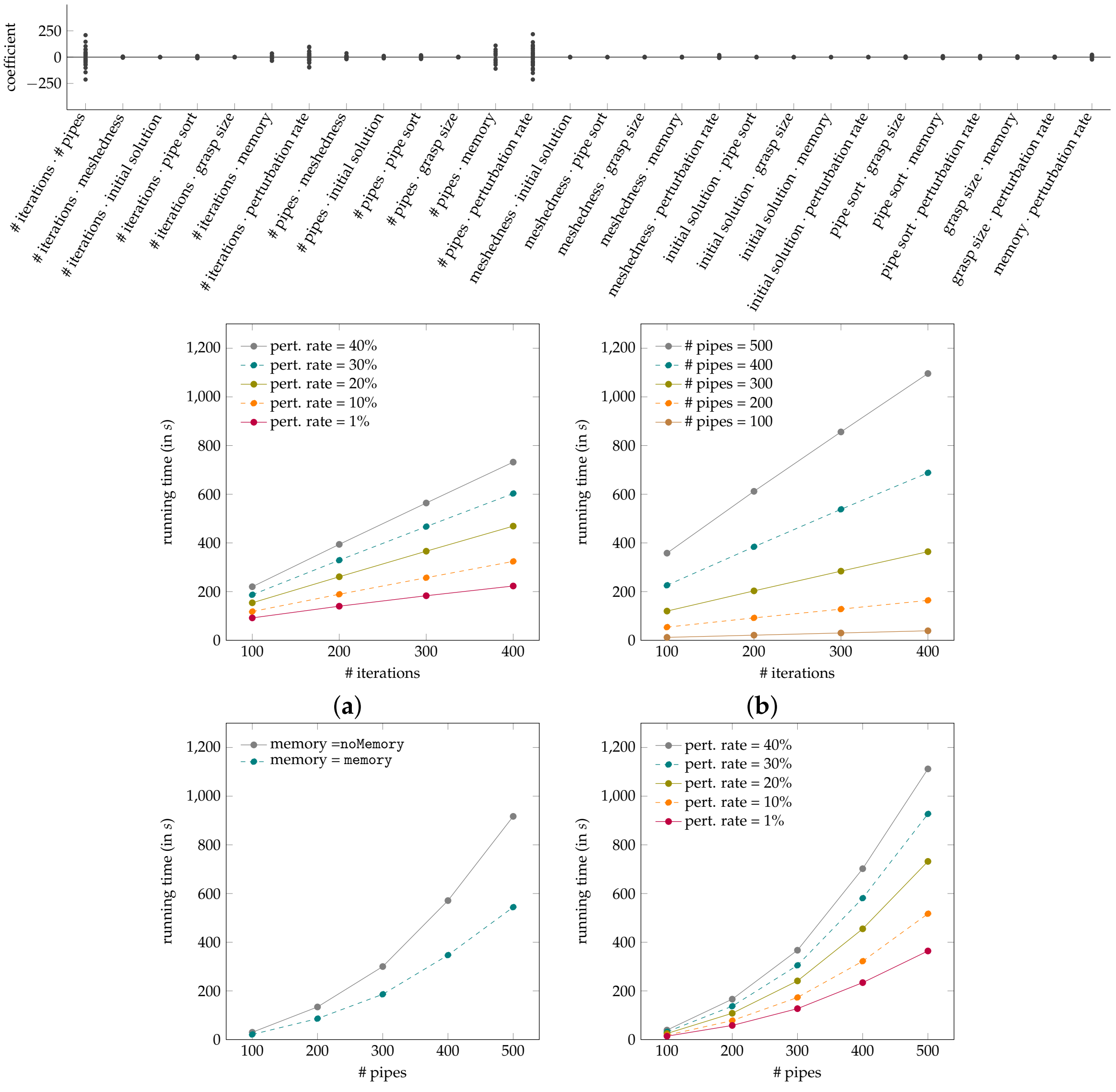


Figure 9. Values of the coefficients and mean plots for the regression model on the *relative cost*: main and interaction effects.



(**a**) (**b**)

(**c**) (**d**)

Figure 10. Values of the coefficients and mean plots for the regression model on the *running time*: main and interaction effects.

3.3.3. Influence of the Different Factors

Initial Solution

The spread of the coefficients depicted in Figure 7 for the model on the relative cost is rather small for this factor. The mean plot in Figure 7b visualizes that the `highCost` strategy generates slightly better results in terms of solution quality. The numerical results also showed that, for most of the test instances, the `lowCost` procedure is not able to construct a feasible "low-cost" solution, and therefore all diameters are set at their maximal value. This implies that most initial solutions generated via the `lowCost` strategy will have a higher cost than those generated under `highCost`.

The manner in which the initial solution is constructed does not influence the algorithm running time, as can be derived from the very small spread of the coefficients in Figure 8. Constructing an initial solution according to the `lowCost` procedure takes some extra seconds compared to `highCost`, but these are negligible compared to the total algorithm running time.

Pipe Sort

Pre-testing showed that sorting the pipes according to decreasing length or according to decreasing cost difference generates much better results in terms of solution quality than randomly sorting the set of pipes. Therefore, this random sorting was not included in the full-factorial experiment, and only sorting according to decreasing length (`lengthSort`) or according to decreasing cost difference (`costSort`) were compared. The very low spread of the coefficients in Figures 7 and 8 shows that there is not much difference between these two ways of sorting. This is logical, since both sorts are related: longer pipes—which will appear at the top of the `lengthSort` list—are more likely to induce higher cost differences, since this cost is a linear function of the pipe length. These longer pipes therefore also appear higher on the list sorted based on decreasing cost differences (`costSort`).

Despite the fact that sorting according to decreasing cost (`costSort`) requires a re-sorting of the list after every local search, there is no significant difference in running time of the two types of sorting, as is visible in Figure 8.

Local Search

The coefficients show that adding a *memory structure* influences both the solution quality and the algorithm running time. From the mean plots in Figures 7e and 8c, it is clear that adding a memory structure to the local search is very beneficial: running times are decreased by 40% without significantly losing solution quality. This is also visualized in Figure 11: the number of calls to EPANET (which is the time-consuming part of the algorithm) is twice as high for the algorithm without memory, whereas the evolution of solution cost is not very different for the algorithm with or without memory structure. Figure 9b shows that the memory structure interacts with the perturbation rate: the algorithm with memory structure is more sensitive to variations in the perturbation rate.

Figure 7d displays that a *grasp size* of 10% leads to the lowest cost solutions. Figure 8 shows that the grasp size does not influence the algorithm running time, nor is there a difference in running time between the algorithm with grasp selection procedure and the algorithm without grasp.

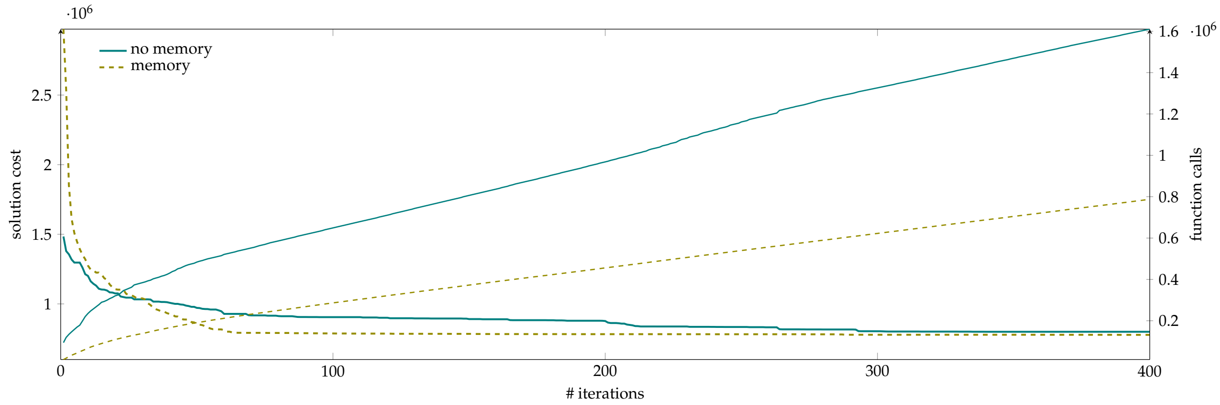


Figure 11. Effect of memory structure.

Perturbation

From the spread of the coefficients in Figures 7 and 8, it is clear that the perturbation rate influences both the solution quality and the algorithm running time. From the mean plot in Figure 7f, it is clear that a perturbation rate of 10% achieves the best results in terms of solution quality. This is explained by the fact that perturbations that are too small are not able to escape the local optima, and on the other hand, perturbation rates that are too high lead to a significant loss of information about the (potentially good) regions of the solution space. As can be seen in Figure 9c, when pipes are sorted according to decreasing length (`lengthSort`), the algorithm is more sensitive to (higher) perturbation rates. When pipes are (re)sorted according to decreasing cost difference, pipes that were perturbed—and thus increased in diameter—will appear relatively higher on the list: this increased diameter implies a higher cost difference that can be realized by decreasing this diameter again. Therefore, especially for higher perturbation rates, a large part of the perturbed pipes will be decreased first, which undoes part of the larger perturbation and moves the current solution back to good areas of the solution space. When pipes are sorted according to decreasing length, the local search moves differently through the solution space and it is less capable of fully exploiting the good areas when larger perturbation rates are used.

Figure 8d visualizes how the running time increases linearly as the perturbation rate increases. There are two reasons for this. Firstly, every change in diameter creates a new solution that has to be simulated hydraulically, since only feasible solutions are allowed as a result of the perturbation. As a consequence, changing more pipes—due to a higher perturbation rate—will take more time. Therefore, the perturbation phase itself will take longer for higher perturbation rates. This is clearly demonstrated in Figure 12a–c. These figures plot 26 iterations of the ILS algorithm executed on a toy network. At the end of every local search phase, a perturbation is performed. It is clear that the horizontal distance between the consecutive local searches is bigger for the perturbation rate of 40%, which implies that the related perturbations are more time consuming. A second reason for the longer running time is that a higher perturbation rate results in solutions that have more "potential" for improvement, which is depicted by the larger vertical distance between two consecutive local search phases. In Figure 12c, it is clear that the perturbation rate of 40% implies more moves that are executed per iteration, or more "new" current solutions that are generated per iteration.

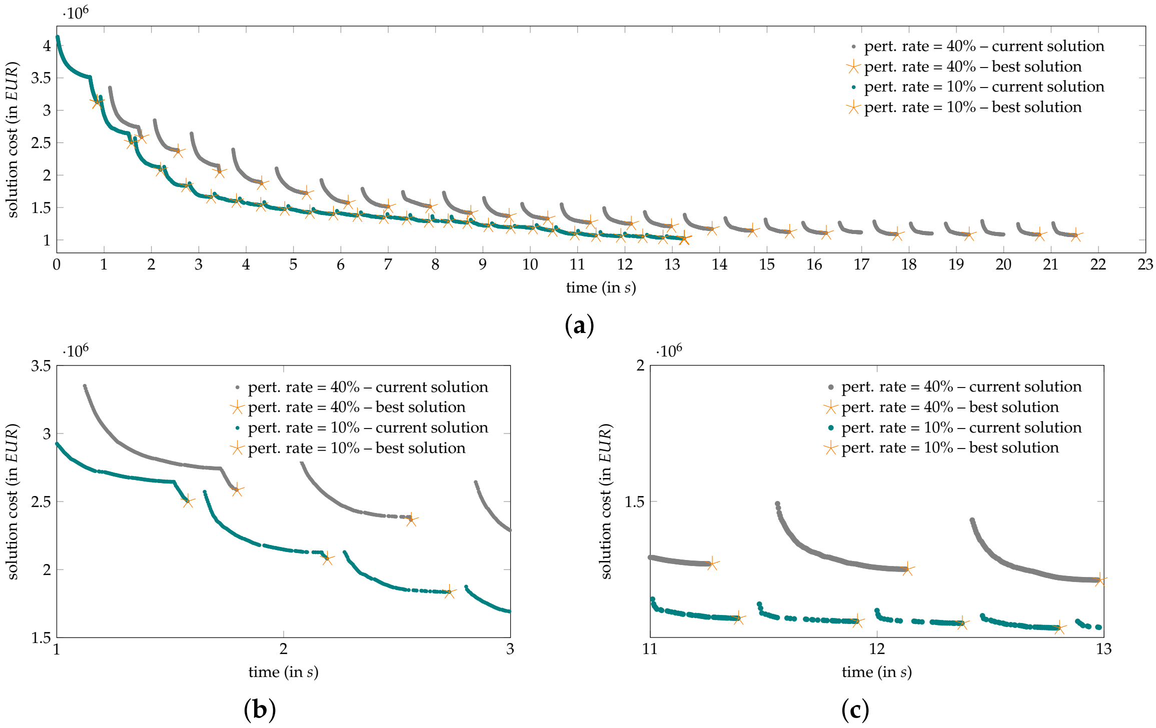


Figure 12. Effect of different perturbation rates.

Termination Criterion

The solution quality is strongly influenced by the maximal number of iterations. It is logical that the solution cost decreases as the number of iterations is increased. The mean plot of Figure 7a shows that the related slope flattens out when the number of iterations increases, which means that most of the improvement is made in the beginning. This is also clearly visible in Figure 13, where the logarithm of the solution cost is plotted as a function of the number of iterations. Even for the bigger networks, the algorithm stabilizes after a relatively small number of iterations, and most of the improvement in solution quality is made at the very beginning. The improvement that is made after a higher number of iterations is negligible compared to the improvement made in the beginning.

The running time increases linearly as a function of the maximal number of iterations, which is to be expected: every iteration follows the same procedure, and will therefore take an approximately equal amount of time. From Figure 10a, it is clear that the number of iterations interacts with the perturbation rate. Lower perturbation rates imply shorter running times. Therefore, the slope of the observations with low perturbation rates will be less steep than the slope of the observations with high perturbation rates. A similar conclusion can be drawn for the interaction between the maximal number of iterations and the network size (Figure 10b).

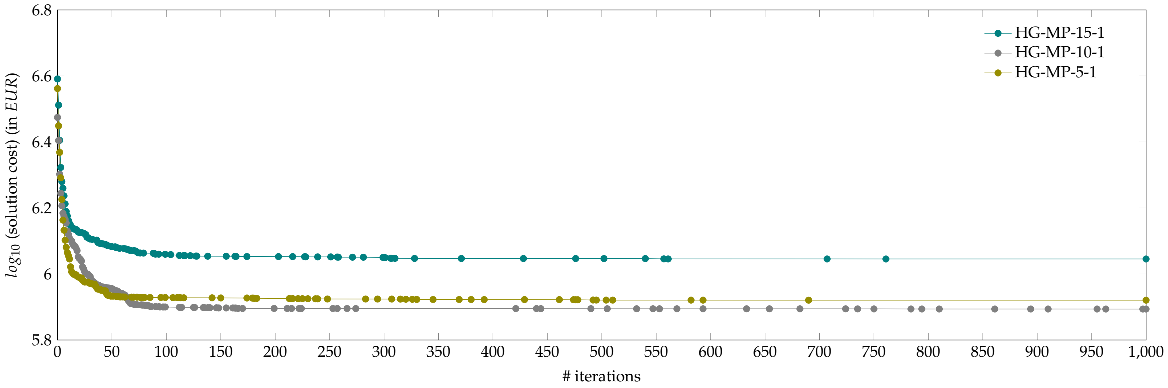


Figure 13. Long run of 1000 iterations for three networks.

Network Size and Structure

It is clear that the number of pipes strongly influences the algorithm running time: larger networks will have longer running times. On the contrary, the meshedness of the network does not significantly influence the algorithm running time. This implies that more looped networks do not necessarily involve longer hydraulic simulation times.

4. Experimental Results

As mentioned above, other authors also tackled similar extensions. Farmani et al. [25] also use time varying demand patterns, but formulate the problem as a multi-objective optimization problem (in contrast to this single-objective formulation). Gupta et al. [19] and Bragalli et al. [26] also impose velocity constraints, but in a single-period setting. Therefore, comparison of results has no use. Since, to the authors' knowledge, no multi-period, single-objective, pressure- and velocity-constrained test instances are available, it is impossible to compare our results to other findings.

Multiple authors (e.g., Maier et al. [32]) have stated that algorithms for WDND optimization should be tested on more, larger, and more challenging instances. Therefore, the ILS algorithm is run on a diverse set of HydroGen [27] test instances, available via [33]. These networks have varying size and characteristics, which enables more robust conclusions regarding the performance of the ILS algorithms to be drawn.

Since none of the previously developed algorithms for the WDND optimization problem are freely available, the results of our algorithm could not be compared to those of other algorithms. Moreover, a re-implementation of these algorithms comes with its own issues, especially the fact that the performance of the re-implemented algorithms might suffer from poor algorithm configuration and calibration. As stressed by Marchi et al. [23], a good calibration of all algorithms is essential in comparative analysis. All test networks, however, are freely available online, and we invite other researchers to test their algorithms on the networks, using their own calibration.

The input files are available via [33]. Information on the instances is given in Table 3.

Table 3. Information on the HydroGen instances.

Network	Meshedness Coefficient	Pipes	Demand Nodes	Water Reservoirs	Network	Meshedness Coefficient	Pipes	Demand Nodes	Water Reservoirs
HG-MP-1	0.20	100	73	1	HG-MP-16	0.20	606	431	4
HG-MP-2	0.15	100	78	1	HG-MP-17	0.15	607	465	4
HG-MP-3	0.10	99	83	1	HG-MP-18	0.10	608	503	5
HG-MP-4	0.20	198	143	1	HG-MP-19	0.20	708	503	5
HG-MP-5	0.15	200	155	1	HG-MP-20	0.15	703	538	5
HG-MP-6	0.10	198	166	1	HG-MP-21	0.10	707	586	5
HG-MP-7	0.20	299	215	2	HG-MP-22	0.20	805	572	6
HG-MP-8	0.15	300	232	2	HG-MP-23	0.15	804	615	6
HG-MP-9	0.10	295	247	2	HG-MP-24	0.10	808	669	6
HG-MP-10	0.20	397	285	2	HG-MP-25	0.20	906	644	6
HG-MP-11	0.15	399	308	2	HG-MP-26	0.15	905	692	7
HG-MP-12	0.10	395	330	3	HG-MP-27	0.10	908	752	7
HG-MP-13	0.20	498	357	2	HG-MP-28	0.20	1,008	716	7
HG-MP-14	0.15	499	385	3	HG-MP-29	0.15	1,007	770	7
HG-MP-15	0.10	495	413	3	HG-MP-30	0.10	1,009	835	8

A set of 16 different pipe types is used. The set of available pipe types and their corresponding costs can be found in Table 4.

Table 4. Available pipe types and their corresponding costs.

Number	Diameter (in mm)	Roughness (unitless)	Cost (in *EUR* per m)	Number	Diameter (in mm)	Roughness (unitless)	Cost (in *EUR* per m)
1	20	130	9	200	130	116	
2	30	130	20	10	250	130	150
3	40	130	25	11	300	130	201
4	50	130	30	12	350	130	246
5	60	130	35	13	400	130	290
6	80	130	38	14	500	130	351
7	100	130	50	15	600	130	528
8	150	130	61	16	1,000	130	628

A minimal pressure head of 20 m is required in every demand node at every time period, and a maximal velocity of 2 m $\cdot$ s^{-1} is set for the water flow through every pipe in the network, at every time period. Demand nodes are divided in five categories (domestic, industrial, energy, public services, and commercial demand nodes), each with a corresponding base load and demand pattern. The base loads can be found in the EPANET input files of the instances. Demand patterns are visualized in Figure 14.

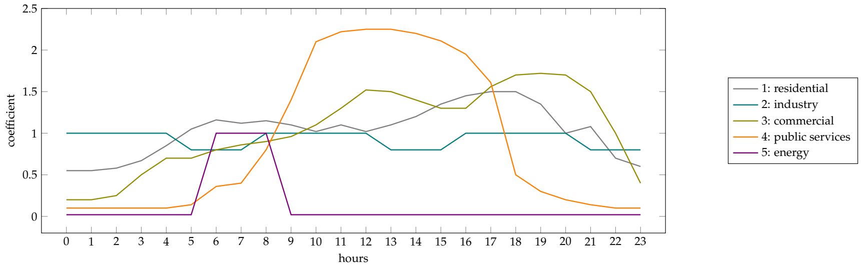


Figure 14. One-day hourly demand patterns of the predefined user categories.

The algorithm was configured with the initial solution constructed according to `highCost`, since this leads to lower cost solutions without negatively influencing the running time. Pipes were sorted according to decreasing length (`lengthSort`), since this yields slightly better results for a perturbation rate of 10%. The grasp size was set at 10%, again, because this leads to lower cost solutions without worsening the running time. The memory structure in the local search is activated to reduce running times, and the perturbation rate is set at 10% since this leads to good solutions in relatively short running times. The algorithm was executed 10 times per instance for a fixed number of calls to EPANET. The lowest cost solution found after a fixed number of function evaluations is recorded, and minimum, mean, and maximum costs over the 10 runs are calculated and reported. These results, together with the mean number of iterations and the standard deviations, can be found in the tables in Appendix A.

From these tables, it is clear that that more iterations are performed on the smaller instances for an equal amount of calls to EPANET. This is to be expected: larger instances imply larger neighborhoods, and therefore, more calls to EPANET (or function evaluations) are required per iteration. For instance, in HG-MP-30-1, only five iterations were finished after 100,000 function evaluations; whereas for HG-MP-1-1, 160 iterations were finished on average.

The minimum, mean, and maximum cost are monotonically decreasing, which is to be expected because the reported results are related to the lowest cost solutions found so far. For some instances, it might seem that these costs do not decrease any further as the number of function calls increases. When looking at the non-rounded results, however, it is clear that the solutions are still improving, but this improvement is very small and therefore non-noticeable from the rounded results.

Another thing to notice is that the percentage decrease of the minimal, mean, and maximal cost is smaller for smaller instances. This is due to the fact that for these smaller instances, more iterations were performed, the algorithm has converged more, and only small additional improvements will be made.

From these tables, it is also clear that it would be interesting to develop a multi-start version of the ILS algorithm. When looking at a small instance (e.g., HG-MP-2-5) it is clear that the minimum and the mean cost found after 100,000 function evaluations are considerably lower than the maximum cost found after 1,500,000 function evaluations. This implies that running the algorithm 15 times for 100,000 function evaluations will almost certainly lead to a lower cost solution than running the algorithm once for 1,500,000 iterations. For bigger instances (e.g., HG-MP-26-5), this conclusion cannot be drawn, at least not for the considered number of evaluations. If the bigger instances were run for a higher number of function evaluations, similar behavior could occur and the (potential) benefit of a multi-start algorithm could appear.

The last column reports the standard deviation of the mean, and gives information on how widely the values are dispersed from this mean. Standard deviations are higher for bigger instances, due to the differences in order of magnitude of the total network cost. Apart from some exceptions (i.e., all smaller instances that already converged), the standard deviation decreases as the number of function evaluations increases. This implies that the ten different runs converge as the algorithm runs longer.

5. Conclusions

The main contribution of this paper is an ILS algorithm that solves the multi-period, velocity-constrained WDND optimization problem quickly and effectively.

The problem is formulated in line with previous research in this area, and aims at finding the optimal pipe configuration out of a set of discrete pipe types, while satisfying hydraulic constraints and customer requirements. This formulation is extended in two ways: (1) an additional constraint—which imposes a maximal water velocity in the distribution pipes—is added, and (2) the problem is extended to a multi-period setting. A simple and easy-to-understand iterated local search algorithm is developed to tackle this mixed-integer, non-linear optimization problem. The developed approach combines a local search step with a perturbation step in order to create a balance between solution space exploitation and exploration, respectively. Only components that show an added value are included in the algorithm. A full-factorial experiment is conducted to test the added value of each of the algorithms' components and to decide on optimal parameter settings. Furthermore, the algorithm is run on a broad set of realistic instances. These are available online and can be used as new benchmark instances and will foster further research in this area.

Other challenges and opportunities for further research remain. An interesting extension of this work would be to extend the gravity-fed formulation to networks that are fed by pumps. Another possible extension is to convert the single-objective formulation to a multi-objective one, where other objectives, such as water quality and network reliability, are also taken into consideration when optimizing network design.

Acknowledgments: This research is funded by the Research Foundation Flanders (FWO).

Author Contributions: Annelies De Corte and Kenneth Sörensen designed the algorithm. Annelies De Corte programmed the algorithm. Annelies De Corte performed the experiments. Annelies De Corte and Kenneth Sörensen analyzed the data. Annelies De Corte wrote the paper and Kenneth Sörensen provided general guidance and reviewed the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Results of ILS on HydroGen Instances

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-1-1	100,000	160	0.381	0.387	0.395	0.004
HG-MP-1-1	300,000	574	0.380	0.385	0.391	0.003
HG-MP-1-1	500,000	988	0.380	0.385	0.391	0.003
HG-MP-1-1	1,000,000	2026	0.380	0.385	0.391	0.003
HG-MP-1-1	1,500,000	3065	0.380	0.385	0.391	0.003
HG-MP-1-2	100,000	155	0.348	0.361	0.414	0.019
HG-MP-1-2	300,000	548	0.347	0.359	0.413	0.020
HG-MP-1-2	500,000	938	0.347	0.359	0.413	0.020
HG-MP-1-2	1,000,000	1904	0.347	0.359	0.413	0.020
HG-MP-1-2	1,500,000	2871	0.346	0.359	0.413	0.020
HG-MP-1-3	100,000	125	0.337	0.343	0.350	0.003
HG-MP-1-3	300,000	482	0.335	0.339	0.342	0.002
HG-MP-1-3	500,000	845	0.334	0.337	0.340	0.002
HG-MP-1-3	1,000,000	1769	0.334	0.336	0.339	0.002
HG-MP-1-3	1,500,000	2695	0.334	0.336	0.339	0.002
HG-MP-1-4	100,000	161	0.340	0.355	0.382	0.011
HG-MP-1-4	300,000	568	0.338	0.353	0.372	0.011
HG-MP-1-4	500,000	974	0.338	0.352	0.366	0.009
HG-MP-1-4	1,000,000	1997	0.338	0.352	0.365	0.009
HG-MP-1-4	1,500,000	3017	0.338	0.351	0.365	0.009
HG-MP-1-5	100,000	160	0.303	0.312	0.318	0.004
HG-MP-1-5	300,000	547	0.298	0.310	0.318	0.006
HG-MP-1-5	500,000	935	0.298	0.310	0.318	0.006
HG-MP-1-5	1,000,000	1902	0.298	0.309	0.318	0.005
HG-MP-1-5	1,500,000	2871	0.298	0.309	0.318	0.005
HG-MP-2-1	100,000	149	0.319	0.395	0.441	0.036
HG-MP-2-1	300,000	530	0.318	0.392	0.437	0.036
HG-MP-2-1	500,000	917	0.318	0.388	0.431	0.034
HG-MP-2-1	1,000,000	1881	0.318	0.387	0.431	0.034
HG-MP-2-1	1,500,000	2849	0.318	0.387	0.431	0.034
HG-MP-2-2	100,000	147	0.289	0.297	0.308	0.006
HG-MP-2-2	300,000	521	0.289	0.294	0.305	0.005
HG-MP-2-2	500,000	895	0.289	0.294	0.305	0.005
HG-MP-2-2	1,000,000	1835	0.289	0.294	0.302	0.004
HG-MP-2-2	1,500,000	2776	0.289	0.294	0.302	0.004
HG-MP-2-3	100,000	150	0.248	0.286	0.313	0.022
HG-MP-2-3	300,000	542	0.245	0.275	0.313	0.025
HG-MP-2-3	500,000	943	0.245	0.274	0.313	0.025
HG-MP-2-3	1,000,000	1953	0.245	0.274	0.313	0.025
HG-MP-2-3	1,500,000	2959	0.245	0.269	0.313	0.023
HG-MP-2-4	100,000	150	0.370	0.398	0.441	0.024
HG-MP-2-4	300,000	544	0.369	0.390	0.437	0.023
HG-MP-2-4	500,000	939	0.362	0.387	0.435	0.023
HG-MP-2-4	1,000,000	1930	0.362	0.386	0.435	0.023
HG-MP-2-4	1,500,000	2923	0.362	0.386	0.435	0.023
HG-MP-2-5	100,000	139	0.349	0.387	0.444	0.043
HG-MP-2-5	300,000	476	0.306	0.381	0.444	0.048
HG-MP-2-5	500,000	814	0.306	0.381	0.444	0.048
HG-MP-2-5	1,000,000	1655	0.306	0.379	0.440	0.048
HG-MP-2-5	1,500,000	2497	0.305	0.374	0.438	0.052

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-3-1	100,000	147	0.449	0.455	0.465	0.005
HG-MP-3-1	300,000	513	0.446	0.451	0.459	0.004
HG-MP-3-1	500,000	878	0.445	0.451	0.459	0.004
HG-MP-3-1	1,000,000	1793	0.445	0.451	0.459	0.004
HG-MP-3-1	1,500,000	2709	0.445	0.451	0.459	0.004
HG-MP-3-2	100,000	141	0.321	0.367	0.431	0.049
HG-MP-3-2	300,000	500	0.318	0.356	0.431	0.048
HG-MP-3-2	500,000	866	0.318	0.355	0.431	0.048
HG-MP-3-2	1,000,000	1785	0.318	0.355	0.431	0.048
HG-MP-3-2	1,500,000	2707	0.318	0.354	0.431	0.048
HG-MP-3-3	100,000	148	0.399	0.419	0.438	0.013
HG-MP-3-3	300,000	536	0.398	0.417	0.437	0.013
HG-MP-3-3	500,000	927	0.398	0.417	0.437	0.013
HG-MP-3-3	1,000,000	1901	0.398	0.416	0.437	0.012
HG-MP-3-3	1,500,000	2876	0.398	0.416	0.437	0.012
HG-MP-3-4	100,000	135	0.343	0.351	0.357	0.004
HG-MP-3-4	300,000	460	0.343	0.350	0.355	0.003
HG-MP-3-4	500,000	786	0.343	0.349	0.355	0.004
HG-MP-3-4	1,000,000	1605	0.343	0.349	0.353	0.003
HG-MP-3-4	1,500,000	2427	0.343	0.349	0.353	0.003
HG-MP-3-5	100,000	143	0.454	0.504	0.555	0.034
HG-MP-3-5	300,000	521	0.442	0.484	0.555	0.041
HG-MP-3-5	500,000	904	0.442	0.472	0.555	0.035
HG-MP-3-5	1,000,000	1865	0.442	0.471	0.555	0.035
HG-MP-3-5	1,500,000	2828	0.442	0.468	0.555	0.033
HG-MP-4-1	100,000	47	0.810	0.855	0.903	0.029
HG-MP-4-1	300,000	249	0.758	0.788	0.845	0.025
HG-MP-4-1	500,000	461	0.755	0.781	0.842	0.027
HG-MP-4-1	1,000,000	996	0.750	0.777	0.841	0.027
HG-MP-4-1	1,500,000	1533	0.750	0.775	0.824	0.023
HG-MP-4-2	100,000	46	0.813	0.866	0.956	0.037
HG-MP-4-2	300,000	229	0.773	0.814	0.854	0.024
HG-MP-4-2	500,000	420	0.770	0.810	0.849	0.024
HG-MP-4-2	1,000,000	901	0.761	0.809	0.849	0.026
HG-MP-4-2	1,500,000	1383	0.761	0.807	0.843	0.025
HG-MP-4-3	100,000	45	0.709	0.736	0.776	0.023
HG-MP-4-3	300,000	238	0.652	0.697	0.742	0.024
HG-MP-4-3	500,000	443	0.642	0.690	0.732	0.026
HG-MP-4-3	1,000,000	957	0.641	0.686	0.730	0.025
HG-MP-4-3	1,500,000	1473	0.640	0.683	0.729	0.023
HG-MP-4-4	100,000	47	0.781	0.824	0.887	0.033
HG-MP-4-4	300,000	250	0.729	0.762	0.801	0.024
HG-MP-4-4	500,000	463	0.725	0.759	0.796	0.024
HG-MP-4-4	1,000,000	998	0.722	0.755	0.795	0.022
HG-MP-4-4	1,500,000	1534	0.722	0.754	0.795	0.023
HG-MP-4-5	100,000	47	0.649	0.677	0.707	0.018
HG-MP-4-5	300,000	250	0.603	0.639	0.679	0.023
HG-MP-4-5	500,000	456	0.599	0.631	0.678	0.025
HG-MP-4-5	1,000,000	975	0.598	0.629	0.674	0.025
HG-MP-4-5	1,500,000	1494	0.598	0.626	0.665	0.023

Network	**# Function Evaluations**	**Mean # Iterations**	**Min Cost (in 10^6 EUR)**	**Mean Cost (in 10^6 EUR)**	**Max Cost (in 10^6 EUR)**	**Stddev Cost (in 10^6 EUR)**
HG-MP-5-1	100,000	44	0.867	0.944	1.002	0.039
HG-MP-5-1	300,000	212	0.860	0.910	0.936	0.028
HG-MP-5-1	500,000	384	0.859	0.903	0.935	0.027
HG-MP-5-1	1,000,000	814	0.856	0.899	0.934	0.027
HG-MP-5-1	1,500,000	1244	0.854	0.898	0.934	0.027
HG-MP-5-2	100,000	45	0.669	0.738	0.828	0.050
HG-MP-5-2	300,000	234	0.640	0.677	0.768	0.034
HG-MP-5-2	500,000	432	0.631	0.671	0.768	0.037
HG-MP-5-2	1,000,000	933	0.631	0.665	0.765	0.037
HG-MP-5-2	1,500,000	1432	0.631	0.665	0.765	0.037
HG-MP-5-3	100,000	43	0.876	0.935	0.994	0.032
HG-MP-5-3	300,000	218	0.826	0.882	0.917	0.030
HG-MP-5-3	500,000	399	0.825	0.872	0.905	0.030
HG-MP-5-3	1,000,000	854	0.817	0.861	0.905	0.029
HG-MP-5-3	1,500,000	1311	0.817	0.859	0.904	0.030
HG-MP-5-4	100,000	47	0.693	0.731	0.785	0.028
HG-MP-5-4	300,000	239	0.664	0.685	0.713	0.014
HG-MP-5-4	500,000	439	0.664	0.680	0.712	0.014
HG-MP-5-4	1,000,000	939	0.664	0.678	0.701	0.011
HG-MP-5-4	1,500,000	1440	0.664	0.678	0.700	0.011
HG-MP-5-5	100,000	46	0.705	0.765	0.842	0.044
HG-MP-5-5	300,000	227	0.678	0.721	0.767	0.031
HG-MP-5-5	500,000	418	0.675	0.720	0.767	0.031
HG-MP-5-5	1,000,000	897	0.675	0.718	0.767	0.032
HG-MP-5-5	1,500,000	1375	0.675	0.717	0.767	0.032
HG-MP-6-1	100,000	44	0.633	0.668	0.712	0.025
HG-MP-6-1	300,000	225	0.624	0.641	0.661	0.014
HG-MP-6-1	500,000	413	0.619	0.636	0.659	0.013
HG-MP-6-1	1,000,000	887	0.618	0.633	0.658	0.012
HG-MP-6-1	1,500,000	1362	0.618	0.632	0.658	0.012
HG-MP-6-2	100,000	47	0.836	0.870	0.896	0.023
HG-MP-6-2	300,000	232	0.809	0.835	0.859	0.018
HG-MP-6-2	500,000	420	0.791	0.828	0.857	0.020
HG-MP-6-2	1,000,000	894	0.773	0.815	0.854	0.023
HG-MP-6-2	1,500,000	1370	0.771	0.814	0.853	0.023
HG-MP-6-3	100,000	47	0.963	0.987	1.043	0.022
HG-MP-6-3	300,000	229	0.905	0.936	0.995	0.026
HG-MP-6-3	500,000	420	0.898	0.923	0.960	0.019
HG-MP-6-3	1,000,000	906	0.894	0.917	0.956	0.018
HG-MP-6-3	1,500,000	1393	0.892	0.915	0.954	0.018
HG-MP-6-4	100,000	46	0.754	0.782	0.855	0.027
HG-MP-6-4	300,000	234	0.719	0.731	0.794	0.021
HG-MP-6-4	500,000	433	0.711	0.722	0.773	0.017
HG-MP-6-4	1,000,000	935	0.705	0.713	0.719	0.004
HG-MP-6-4	1,500,000	1442	0.699	0.711	0.718	0.005
HG-MP-6-5	100,000	47	0.797	0.830	0.904	0.033
HG-MP-6-5	300,000	223	0.775	0.792	0.812	0.014
HG-MP-6-5	500,000	402	0.772	0.789	0.812	0.014
HG-MP-6-5	1,000,000	853	0.772	0.787	0.811	0.014
HG-MP-6-5	1,500,000	1307	0.772	0.784	0.809	0.011

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-7-1	100,000	21	1.189	1.377	1.557	0.126
HG-MP-7-1	300,000	148	0.903	1.080	1.296	0.116
HG-MP-7-1	500,000	291	0.901	1.060	1.291	0.120
HG-MP-7-1	1,000,000	651	0.894	1.033	1.201	0.105
HG-MP-7-1	1,500,000	1011	0.893	1.022	1.200	0.106
HG-MP-7-2	100,000	22	1.032	1.077	1.115	0.028
HG-MP-7-2	300,000	141	0.882	0.964	1.046	0.043
HG-MP-7-2	500,000	271	0.874	0.952	1.027	0.042
HG-MP-7-2	1,000,000	604	0.872	0.943	1.015	0.042
HG-MP-7-2	1,500,000	940	0.858	0.938	1.010	0.045
HG-MP-7-3	100,000	26	0.938	0.996	1.082	0.040
HG-MP-7-3	300,000	141	0.774	0.825	0.890	0.037
HG-MP-7-3	500,000	274	0.765	0.813	0.878	0.031
HG-MP-7-3	1,000,000	611	0.762	0.807	0.875	0.032
HG-MP-7-3	1,500,000	947	0.762	0.807	0.873	0.031
HG-MP-7-4	100,000	23	1.074	1.167	1.276	0.060
HG-MP-7-4	300,000	158	0.947	0.981	1.026	0.020
HG-MP-7-4	500,000	305	0.943	0.975	1.023	0.020
HG-MP-7-4	1,000,000	675	0.927	0.964	1.014	0.022
HG-MP-7-4	1,500,000	1045	0.924	0.962	1.014	0.023
HG-MP-7-5	100,000	24	0.774	0.863	0.976	0.053
HG-MP-7-5	300,000	148	0.656	0.730	0.927	0.074
HG-MP-7-5	500,000	287	0.654	0.714	0.904	0.072
HG-MP-7-5	1,000,000	649	0.654	0.709	0.896	0.070
HG-MP-7-5	1,500,000	1016	0.653	0.696	0.779	0.042
HG-MP-8-1	100,000	22	0.995	1.075	1.191	0.057
HG-MP-8-1	300,000	137	0.823	0.906	0.999	0.063
HG-MP-8-1	500,000	264	0.812	0.892	0.988	0.059
HG-MP-8-1	1,000,000	587	0.810	0.879	0.979	0.054
HG-MP-8-1	1,500,000	914	0.807	0.875	0.975	0.053
HG-MP-8-2	100,000	21	0.942	1.063	1.135	0.057
HG-MP-8-2	300,000	126	0.883	0.953	1.024	0.045
HG-MP-8-2	500,000	237	0.848	0.937	1.009	0.052
HG-MP-8-2	1,000,000	521	0.834	0.919	0.969	0.047
HG-MP-8-2	1,500,000	806	0.833	0.912	0.958	0.043
HG-MP-8-3	100,000	23	0.903	1.013	1.127	0.068
HG-MP-8-3	300,000	144	0.825	0.888	0.970	0.053
HG-MP-8-3	500,000	275	0.820	0.882	0.953	0.051
HG-MP-8-3	1,000,000	606	0.819	0.867	0.947	0.046
HG-MP-8-3	1,500,000	943	0.818	0.866	0.946	0.046
HG-MP-8-4	100,000	22	1.150	1.264	1.344	0.062
HG-MP-8-4	300,000	141	0.891	1.019	1.180	0.081
HG-MP-8-4	500,000	277	0.879	0.986	1.159	0.076
HG-MP-8-4	1,000,000	621	0.868	0.974	1.128	0.073
HG-MP-8-4	1,500,000	966	0.866	0.972	1.123	0.073
HG-MP-8-5	100,000	24	1.226	1.360	1.441	0.061
HG-MP-8-5	300,000	143	0.966	1.121	1.232	0.073
HG-MP-8-5	500,000	276	0.948	1.091	1.185	0.064
HG-MP-8-5	1,000,000	619	0.938	1.072	1.160	0.074
HG-MP-8-5	1,500,000	962	0.934	1.055	1.140	0.070

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-9-1	100,000	21	0.938	1.015	1.107	0.052
HG-MP-9-1	300,000	140	0.867	0.896	0.921	0.019
HG-MP-9-1	500,000	269	0.865	0.887	0.912	0.017
HG-MP-9-1	1,000,000	597	0.854	0.879	0.894	0.015
HG-MP-9-1	1,500,000	926	0.854	0.876	0.891	0.013
HG-MP-9-2	100,000	26	1.083	1.141	1.221	0.035
HG-MP-9-2	300,000	142	0.856	0.911	1.031	0.049
HG-MP-9-2	500,000	272	0.841	0.889	1.009	0.047
HG-MP-9-2	1,000,000	601	0.837	0.879	1.004	0.047
HG-MP-9-2	1,500,000	930	0.837	0.877	1.000	0.046
HG-MP-9-3	100,000	25	0.997	1.021	1.073	0.023
HG-MP-9-3	300,000	143	0.878	0.915	0.971	0.026
HG-MP-9-3	500,000	274	0.856	0.894	0.967	0.029
HG-MP-9-3	1,000,000	607	0.846	0.887	0.960	0.031
HG-MP-9-3	1,500,000	940	0.841	0.885	0.957	0.031
HG-MP-9-4	100,000	24	0.812	0.867	0.933	0.042
HG-MP-9-4	300,000	134	0.735	0.763	0.786	0.019
HG-MP-9-4	500,000	254	0.730	0.752	0.778	0.017
HG-MP-9-4	1,000,000	561	0.727	0.748	0.774	0.017
HG-MP-9-4	1,500,000	871	0.725	0.746	0.771	0.017
HG-MP-9-5	100,000	25	0.979	1.009	1.106	0.034
HG-MP-9-5	300,000	136	0.909	0.941	0.984	0.018
HG-MP-9-5	500,000	252	0.907	0.935	0.965	0.015
HG-MP-9-5	1,000,000	548	0.905	0.919	0.934	0.008
HG-MP-9-5	1,500,000	847	0.899	0.914	0.932	0.009
HG-MP-10-1	100,000	16	1.181	1.260	1.337	0.052
HG-MP-10-1	300,000	99	0.801	0.837	0.893	0.030
HG-MP-10-1	500,000	208	0.779	0.809	0.856	0.023
HG-MP-10-1	1,000,000	493	0.773	0.800	0.850	0.026
HG-MP-10-1	1,500,000	780	0.772	0.798	0.847	0.025
HG-MP-10-2	100,000	17	1.053	1.090	1.173	0.032
HG-MP-10-2	300,000	100	0.851	0.893	0.937	0.026
HG-MP-10-2	500,000	207	0.842	0.881	0.917	0.024
HG-MP-10-2	1,000,000	487	0.830	0.873	0.905	0.025
HG-MP-10-2	1,500,000	773	0.830	0.871	0.902	0.024
HG-MP-10-3	100,000	18	1.083	1.138	1.223	0.042
HG-MP-10-3	300,000	105	0.840	0.889	1.014	0.052
HG-MP-10-3	500,000	223	0.801	0.862	0.978	0.050
HG-MP-10-3	1,000,000	527	0.795	0.851	0.949	0.043
HG-MP-10-3	1,500,000	838	0.793	0.848	0.939	0.041
HG-MP-10-4	100,000	16	0.928	0.985	1.072	0.039
HG-MP-10-4	300,000	90	0.741	0.802	0.861	0.043
HG-MP-10-4	500,000	181	0.729	0.779	0.838	0.036
HG-MP-10-4	1,000,000	428	0.725	0.770	0.837	0.034
HG-MP-10-4	1,500,000	679	0.724	0.768	0.835	0.034
HG-MP-10-5	100,000	18	1.124	1.236	1.372	0.067
HG-MP-10-5	300,000	96	0.862	0.935	1.008	0.041
HG-MP-10-5	500,000	206	0.859	0.899	0.957	0.031
HG-MP-10-5	1,000,000	496	0.835	0.887	0.942	0.032
HG-MP-10-5	1,500,000	792	0.834	0.883	0.926	0.029

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-11-1	100,000	14	1.171	1.249	1.416	0.081
HG-MP-11-1	300,000	100	0.941	0.990	1.084	0.043
HG-MP-11-1	500,000	198	0.934	0.973	1.069	0.043
HG-MP-11-1	1,000,000	452	0.916	0.960	1.049	0.040
HG-MP-11-1	1,500,000	711	0.914	0.958	1.045	0.039
HG-MP-11-2	100,000	13	1.147	1.241	1.431	0.075
HG-MP-11-2	300,000	96	0.995	1.063	1.147	0.044
HG-MP-11-2	500,000	190	0.976	1.035	1.125	0.045
HG-MP-11-2	1,000,000	437	0.955	1.014	1.088	0.044
HG-MP-11-2	1,500,000	688	0.944	1.005	1.083	0.045
HG-MP-11-3	100,000	14	1.257	1.324	1.404	0.040
HG-MP-11-3	300,000	89	0.913	0.991	1.043	0.040
HG-MP-11-3	500,000	174	0.893	0.953	1.005	0.033
HG-MP-11-3	1,000,000	395	0.874	0.939	0.999	0.033
HG-MP-11-3	1,500,000	620	0.872	0.935	0.997	0.034
HG-MP-11-4	100,000	16	1.038	1.149	1.273	0.081
HG-MP-11-4	300,000	103	0.887	0.919	0.966	0.022
HG-MP-11-4	500,000	205	0.875	0.903	0.941	0.018
HG-MP-11-4	1,000,000	465	0.858	0.890	0.929	0.022
HG-MP-11-4	1,500,000	725	0.856	0.885	0.922	0.018
HG-MP-11-5	100,000	15	1.045	1.155	1.263	0.074
HG-MP-11-5	300,000	103	0.958	1.018	1.124	0.043
HG-MP-11-5	500,000	198	0.949	1.010	1.114	0.043
HG-MP-11-5	1,000,000	440	0.949	1.005	1.104	0.041
HG-MP-11-5	1,500,000	685	0.949	1.003	1.099	0.040
HG-MP-12-1	100,000	15	1.414	1.467	1.559	0.046
HG-MP-12-1	300,000	101	1.112	1.178	1.335	0.070
HG-MP-12-1	500,000	199	1.109	1.152	1.261	0.050
HG-MP-12-1	1,000,000	450	1.091	1.139	1.249	0.048
HG-MP-12-1	1,500,000	702	1.078	1.135	1.243	0.049
HG-MP-12-2	100,000	15	1.342	1.425	1.529	0.054
HG-MP-12-2	300,000	92	1.142	1.191	1.257	0.029
HG-MP-12-2	500,000	182	1.127	1.170	1.234	0.029
HG-MP-12-2	1,000,000	414	1.112	1.146	1.204	0.029
HG-MP-12-2	1,500,000	650	1.088	1.138	1.201	0.032
HG-MP-12-3	100,000	17	1.195	1.247	1.299	0.035
HG-MP-12-3	300,000	95	1.071	1.126	1.169	0.035
HG-MP-12-3	500,000	186	1.057	1.100	1.146	0.032
HG-MP-12-3	1,000,000	421	1.033	1.081	1.137	0.034
HG-MP-12-3	1,500,000	661	1.026	1.071	1.133	0.035
HG-MP-12-4	100,000	13	1.178	1.282	1.336	0.045
HG-MP-12-4	300,000	89	1.037	1.085	1.136	0.036
HG-MP-12-4	500,000	172	1.023	1.056	1.098	0.024
HG-MP-12-4	1,000,000	388	1.006	1.035	1.076	0.024
HG-MP-12-4	1,500,000	611	1.002	1.031	1.071	0.024
HG-MP-12-5	100,000	14	1.114	1.249	1.380	0.067
HG-MP-12-5	300,000	108	0.989	1.015	1.043	0.018
HG-MP-12-5	500,000	213	0.972	1.006	1.039	0.020
HG-MP-12-5	1,000,000	478	0.966	0.995	1.019	0.017
HG-MP-12-5	1,500,000	745	0.920	0.988	1.018	0.027

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-13-1	100,000	11	1.987	2.235	2.583	0.169
HG-MP-13-1	300,000	61	1.371	1.457	1.537	0.059
HG-MP-13-1	500,000	129	1.263	1.349	1.440	0.053
HG-MP-13-1	1,000,000	323	1.223	1.285	1.400	0.053
HG-MP-13-1	1,500,000	531	1.207	1.264	1.360	0.045
HG-MP-13-2	100,000	11	1.918	2.066	2.247	0.096
HG-MP-13-2	300,000	66	1.371	1.457	1.610	0.070
HG-MP-13-2	500,000	139	1.304	1.372	1.490	0.048
HG-MP-13-2	1,000,000	339	1.284	1.317	1.358	0.024
HG-MP-13-2	1,500,000	547	1.268	1.301	1.349	0.026
HG-MP-13-3	100,000	10	1.656	1.753	1.817	0.054
HG-MP-13-3	300,000	64	1.173	1.244	1.343	0.048
HG-MP-13-3	500,000	134	1.165	1.212	1.291	0.036
HG-MP-13-3	1,000,000	324	1.131	1.184	1.242	0.028
HG-MP-13-3	1,500,000	515	1.127	1.178	1.232	0.027
HG-MP-13-4	100,000	11	2.010	2.158	2.411	0.133
HG-MP-13-4	300,000	66	1.277	1.338	1.377	0.031
HG-MP-13-4	500,000	142	1.256	1.285	1.318	0.021
HG-MP-13-4	1,000,000	344	1.210	1.251	1.272	0.019
HG-MP-13-4	1,500,000	551	1.205	1.235	1.269	0.019
HG-MP-13-5	100,000	10	1.787	1.897	2.034	0.083
HG-MP-13-5	300,000	67	1.266	1.363	1.498	0.082
HG-MP-13-5	500,000	139	1.191	1.262	1.367	0.047
HG-MP-13-5	1,000,000	338	1.153	1.195	1.275	0.037
HG-MP-13-5	1,500,000	543	1.145	1.183	1.264	0.037
HG-MP-14-1	100,000	9	1.559	1.623	1.727	0.059
HG-MP-14-1	300,000	70	1.041	1.140	1.272	0.075
HG-MP-14-1	500,000	150	1.026	1.102	1.257	0.072
HG-MP-14-1	1,000,000	359	1.020	1.068	1.161	0.051
HG-MP-14-1	1,500,000	573	1.013	1.048	1.117	0.038
HG-MP-14-2	100,000	11	1.916	1.987	2.069	0.055
HG-MP-14-2	300,000	65	1.442	1.585	1.754	0.107
HG-MP-14-2	500,000	132	1.376	1.482	1.678	0.115
HG-MP-14-2	1,000,000	306	1.283	1.359	1.455	0.051
HG-MP-14-2	1,500,000	489	1.272	1.332	1.430	0.042
HG-MP-14-3	100,000	10	1.590	1.649	1.744	0.045
HG-MP-14-3	300,000	68	1.138	1.209	1.313	0.052
HG-MP-14-3	500,000	143	1.127	1.173	1.277	0.040
HG-MP-14-3	1,000,000	339	1.105	1.151	1.269	0.043
HG-MP-14-3	1,500,000	538	1.101	1.143	1.242	0.037
HG-MP-14-4	100,000	10	1.557	1.667	1.812	0.064
HG-MP-14-4	300,000	71	1.131	1.208	1.256	0.038
HG-MP-14-4	500,000	153	1.121	1.174	1.231	0.037
HG-MP-14-4	1,000,000	364	1.096	1.155	1.224	0.045
HG-MP-14-4	1,500,000	580	1.076	1.144	1.223	0.051
HG-MP-14-5	100,000	11	1.267	1.398	1.679	0.118
HG-MP-14-5	300,000	74	0.927	1.002	1.101	0.052
HG-MP-14-5	500,000	156	0.898	0.961	1.043	0.039
HG-MP-14-5	1,000,000	373	0.891	0.949	1.028	0.037
HG-MP-14-5	1,500,000	591	0.889	0.943	1.022	0.035

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-15-1	100,000	11	1.358	1.427	1.479	0.044
HG-MP-15-1	300,000	74	1.159	1.212	1.287	0.037
HG-MP-15-1	500,000	145	1.128	1.181	1.236	0.033
HG-MP-15-1	1,000,000	334	1.085	1.153	1.188	0.030
HG-MP-15-1	1,500,000	528	1.069	1.142	1.179	0.033
HG-MP-15-2	100,000	10	1.814	2.138	2.325	0.152
HG-MP-15-2	300,000	58	1.113	1.277	1.647	0.169
HG-MP-15-2	500,000	121	1.064	1.131	1.362	0.083
HG-MP-15-2	1,000,000	289	1.046	1.095	1.308	0.072
HG-MP-15-2	1,500,000	463	1.039	1.086	1.283	0.068
HG-MP-15-3	100,000	11	1.662	1.816	2.219	0.182
HG-MP-15-3	300,000	71	1.280	1.346	1.414	0.044
HG-MP-15-3	500,000	150	1.222	1.294	1.386	0.049
HG-MP-15-3	1,000,000	357	1.210	1.272	1.348	0.044
HG-MP-15-3	1,500,000	568	1.206	1.264	1.319	0.039
HG-MP-15-4	100,000	11	1.368	1.447	1.620	0.072
HG-MP-15-4	300,000	77	1.039	1.102	1.151	0.032
HG-MP-15-4	500,000	158	1.020	1.069	1.130	0.028
HG-MP-15-4	1,000,000	367	1.018	1.053	1.117	0.025
HG-MP-15-4	1,500,000	577	1.018	1.049	1.117	0.026
HG-MP-15-5	100,000	12	1.720	1.835	1.965	0.083
HG-MP-15-5	300,000	78	1.448	1.479	1.546	0.030
HG-MP-15-5	500,000	158	1.370	1.424	1.468	0.030
HG-MP-15-5	1,000,000	363	1.315	1.394	1.446	0.036
HG-MP-15-5	1,500,000	574	1.281	1.378	1.420	0.040
HG-MP-16-1	100,000	9	2.600	2.700	2.881	0.077
HG-MP-16-1	300,000	51	1.685	1.867	2.156	0.127
HG-MP-16-1	500,000	111	1.487	1.677	1.984	0.170
HG-MP-16-1	1,000,000	275	1.456	1.611	1.889	0.148
HG-MP-16-1	1,500,000	441	1.451	1.585	1.864	0.141
HG-MP-16-2	100,000	9	2.394	2.503	2.609	0.060
HG-MP-16-2	300,000	50	1.729	1.806	1.893	0.052
HG-MP-16-2	500,000	108	1.612	1.687	1.770	0.049
HG-MP-16-2	1,000,000	267	1.557	1.633	1.697	0.045
HG-MP-16-2	1,500,000	431	1.551	1.620	1.676	0.041
HG-MP-16-3	100,000	8	2.362	2.446	2.560	0.066
HG-MP-16-3	300,000	53	1.548	1.665	1.809	0.086
HG-MP-16-3	500,000	119	1.464	1.558	1.638	0.051
HG-MP-16-3	1,000,000	295	1.453	1.518	1.565	0.032
HG-MP-16-3	1,500,000	477	1.440	1.498	1.538	0.030
HG-MP-16-4	100,000	9	2.272	2.379	2.592	0.105
HG-MP-16-4	300,000	52	1.367	1.546	1.726	0.114
HG-MP-16-4	500,000	112	1.309	1.485	1.673	0.109
HG-MP-16-4	1,000,000	273	1.278	1.420	1.596	0.101
HG-MP-16-4	1,500,000	437	1.273	1.409	1.591	0.104
HG-MP-16-5	100,000	9	2.968	3.128	3.268	0.096
HG-MP-16-5	300,000	51	1.928	2.077	2.232	0.085
HG-MP-16-5	500,000	112	1.815	1.893	1.977	0.051
HG-MP-16-5	1,000,000	276	1.641	1.818	1.952	0.088
HG-MP-16-5	1,500,000	444	1.626	1.804	1.929	0.086

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-17-1	100,000	9	2.557	2.829	3.053	0.133
HG-MP-17-1	300,000	54	1.749	1.899	2.048	0.094
HG-MP-17-1	500,000	117	1.571	1.796	1.951	0.099
HG-MP-17-1	1,000,000	288	1.402	1.720	1.893	0.139
HG-MP-17-1	1,500,000	463	1.400	1.704	1.870	0.137
HG-MP-17-2	100,000	8	2.727	2.873	3.186	0.141
HG-MP-17-2	300,000	49	1.846	1.918	2.024	0.050
HG-MP-17-2	500,000	106	1.686	1.784	1.851	0.049
HG-MP-17-2	1,000,000	260	1.619	1.728	1.818	0.056
HG-MP-17-2	1,500,000	418	1.611	1.715	1.798	0.054
HG-MP-17-3	100,000	9	2.318	2.605	2.927	0.160
HG-MP-17-3	300,000	46	1.592	1.761	1.894	0.090
HG-MP-17-3	500,000	99	1.530	1.632	1.695	0.054
HG-MP-17-3	1,000,000	243	1.481	1.566	1.652	0.052
HG-MP-17-3	1,500,000	391	1.444	1.533	1.638	0.057
HG-MP-17-4	100,000	8	2.761	2.939	3.147	0.104
HG-MP-17-4	300,000	51	1.641	1.830	1.905	0.080
HG-MP-17-4	500,000	115	1.614	1.771	1.851	0.071
HG-MP-17-4	1,000,000	286	1.611	1.734	1.820	0.064
HG-MP-17-4	1,500,000	461	1.606	1.725	1.814	0.065
HG-MP-17-5	100,000	8	3.594	3.744	3.968	0.127
HG-MP-17-5	300,000	49	1.951	2.142	2.411	0.126
HG-MP-17-5	500,000	117	1.817	1.983	2.176	0.112
HG-MP-17-5	1,000,000	299	1.759	1.908	2.062	0.099
HG-MP-17-5	1,500,000	483	1.749	1.890	2.053	0.088
HG-MP-18-1	100,000	9	2.378	2.513	2.744	0.131
HG-MP-18-1	300,000	54	1.416	1.637	1.800	0.138
HG-MP-18-1	500,000	115	1.311	1.529	1.703	0.127
HG-MP-18-1	1,000,000	273	1.288	1.505	1.661	0.120
HG-MP-18-1	1,500,000	433	1.284	1.498	1.655	0.119
HG-MP-18-2	100,000	7	2.304	2.513	2.643	0.102
HG-MP-18-2	300,000	46	1.322	1.444	1.566	0.069
HG-MP-18-2	500,000	105	1.205	1.284	1.369	0.047
HG-MP-18-2	1,000,000	264	1.173	1.238	1.312	0.044
HG-MP-18-2	1,500,000	431	1.167	1.229	1.301	0.043
HG-MP-18-3	100,000	8	2.955	3.084	3.182	0.069
HG-MP-18-3	300,000	48	1.958	2.054	2.225	0.089
HG-MP-18-3	500,000	108	1.848	1.934	2.098	0.064
HG-MP-18-3	1,000,000	267	1.780	1.865	2.014	0.060
HG-MP-18-3	1,500,000	429	1.760	1.845	2.000	0.060
HG-MP-18-4	100,000	8	3.027	3.234	3.487	0.152
HG-MP-18-4	300,000	48	1.827	1.969	2.261	0.111
HG-MP-18-4	500,000	109	1.750	1.867	2.072	0.089
HG-MP-18-4	1,000,000	278	1.686	1.813	1.976	0.076
HG-MP-18-4	1,500,000	451	1.675	1.797	1.956	0.074
HG-MP-18-5	100,000	8	2.714	2.920	3.291	0.197
HG-MP-18-5	300,000	44	1.681	1.832	2.097	0.125
HG-MP-18-5	500,000	99	1.624	1.705	1.813	0.060
HG-MP-18-5	1,000,000	243	1.567	1.621	1.693	0.038
HG-MP-18-5	1,500,000	392	1.563	1.605	1.683	0.038

Network	# Function Evaluations	Mean # Iterations	Min Cost	Mean Cost	Max Cost	Stddev Cost
			(in 10^6 EUR)	(in 10^6 EUR)	(in 10^6 EUR)	(in 10^6 EUR)
HG-MP-19-1	100,000	8	4.045	4.323	4.553	0.158
HG-MP-19-1	300,000	40	2.553	2.694	2.801	0.076
HG-MP-19-1	500,000	90	2.267	2.459	2.602	0.085
HG-MP-19-1	1,000,000	233	2.194	2.369	2.531	0.095
HG-MP-19-1	1,500,000	382	2.183	2.346	2.514	0.094
HG-MP-19-2	100,000	7	4.696	5.144	5.656	0.269
HG-MP-19-2	300,000	37	2.966	3.250	3.491	0.146
HG-MP-19-2	500,000	82	2.689	2.857	3.411	0.220
HG-MP-19-2	1,000,000	220	2.423	2.610	2.998	0.151
HG-MP-19-2	1,500,000	366	2.386	2.537	2.905	0.141
HG-MP-19-3	100,000	7	3.472	3.763	4.377	0.244
HG-MP-19-3	300,000	41	2.327	2.400	2.496	0.049
HG-MP-19-3	500,000	93	2.039	2.214	2.343	0.086
HG-MP-19-3	1,000,000	235	1.973	2.124	2.242	0.104
HG-MP-19-3	1,500,000	383	1.946	2.110	2.232	0.106
HG-MP-19-4	100,000	7	3.132	3.578	4.251	0.341
HG-MP-19-4	300,000	38	1.920	2.073	2.231	0.094
HG-MP-19-4	500,000	89	1.837	1.942	2.077	0.078
HG-MP-19-4	1,000,000	227	1.802	1.883	2.033	0.070
HG-MP-19-4	1,500,000	367	1.761	1.846	1.938	0.054
HG-MP-19-5	100,000	8	4.624	4.822	5.025	0.128
HG-MP-19-5	300,000	37	2.740	3.008	3.238	0.173
HG-MP-19-5	500,000	86	2.566	2.721	3.068	0.147
HG-MP-19-5	1,000,000	225	2.413	2.580	2.984	0.157
HG-MP-19-5	1,500,000	370	2.385	2.553	2.922	0.151
HG-MP-20-1	100,000	7	6.054	6.502	7.171	0.406
HG-MP-20-1	300,000	29	2.624	2.896	3.197	0.161
HG-MP-20-1	500,000	64	1.951	2.338	2.534	0.189
HG-MP-20-1	1,000,000	177	1.863	2.106	2.387	0.153
HG-MP-20-1	1,500,000	298	1.815	2.079	2.353	0.158
HG-MP-20-2	100,000	7	2.957	3.075	3.164	0.066
HG-MP-20-2	300,000	44	2.011	2.115	2.213	0.059
HG-MP-20-2	500,000	98	1.889	1.996	2.138	0.078
HG-MP-20-2	1,000,000	249	1.755	1.922	2.065	0.086
HG-MP-20-2	1,500,000	406	1.732	1.898	2.063	0.089
HG-MP-20-3	100,000	7	3.866	4.021	4.160	0.092
HG-MP-20-3	300,000	37	2.393	2.555	2.654	0.083
HG-MP-20-3	500,000	82	2.281	2.442	2.583	0.096
HG-MP-20-3	1,000,000	210	2.195	2.295	2.406	0.068
HG-MP-20-3	1,500,000	347	2.191	2.264	2.375	0.057
HG-MP-20-4	100,000	7	4.005	4.359	4.829	0.231
HG-MP-20-4	300,000	39	2.576	2.848	3.328	0.264
HG-MP-20-4	500,000	88	2.266	2.635	3.189	0.287
HG-MP-20-4	1,000,000	229	2.114	2.279	2.572	0.118
HG-MP-20-4	1,500,000	383	2.080	2.209	2.412	0.097
HG-MP-20-5	100,000	8	4.392	4.575	4.767	0.107
HG-MP-20-5	300,000	38	2.848	3.057	3.197	0.103
HG-MP-20-5	500,000	85	2.355	2.641	2.962	0.170
HG-MP-20-5	1,000,000	225	2.241	2.449	2.787	0.156
HG-MP-20-5	1,500,000	371	2.218	2.416	2.758	0.161

Network	# Function Evaluations	Mean # Iterations	Min Cost	Mean Cost	Max Cost	Stddev Cost
			(in 10^6 EUR)	(in 10^6 EUR)	(in 10^6 EUR)	(in 10^6 EUR)
HG-MP-21-1	100,000	7	4.193	4.443	4.693	0.175
HG-MP-21-1	300,000	36	2.283	2.694	2.926	0.198
HG-MP-21-1	500,000	88	2.187	2.439	2.868	0.212
HG-MP-21-1	1,000,000	232	2.069	2.278	2.648	0.189
HG-MP-21-1	1,500,000	381	2.053	2.234	2.606	0.166
HG-MP-21-2	100,000	7	4.089	4.551	4.738	0.184
HG-MP-21-2	300,000	36	2.597	2.850	3.123	0.182
HG-MP-21-2	500,000	85	2.423	2.526	2.809	0.117
HG-MP-21-2	1,000,000	218	2.283	2.385	2.552	0.078
HG-MP-21-2	1,500,000	357	2.258	2.363	2.506	0.073
HG-MP-21-3	100,000	7	3.576	3.738	4.005	0.132
HG-MP-21-3	300,000	38	2.359	2.511	2.732	0.110
HG-MP-21-3	500,000	84	2.242	2.362	2.489	0.080
HG-MP-21-3	1,000,000	211	2.150	2.287	2.419	0.078
HG-MP-21-3	1,500,000	343	2.132	2.256	2.387	0.073
HG-MP-21-4	100,000	7	4.035	4.421	4.864	0.229
HG-MP-21-4	300,000	41	2.571	2.760	3.156	0.180
HG-MP-21-4	500,000	91	2.430	2.584	3.027	0.172
HG-MP-21-4	1,000,000	228	2.358	2.445	2.648	0.075
HG-MP-21-4	1,500,000	371	2.326	2.418	2.569	0.061
HG-MP-21-5	100,000	7	3.394	3.624	3.970	0.154
HG-MP-21-5	300,000	36	1.934	2.090	2.285	0.110
HG-MP-21-5	500,000	77	1.852	1.928	1.987	0.049
HG-MP-21-5	1,000,000	187	1.811	1.871	1.932	0.041
HG-MP-21-5	1,500,000	299	1.807	1.849	1.912	0.035
HG-MP-22-1	100,000	6	4.360	4.724	4.947	0.192
HG-MP-22-1	300,000	29	2.653	2.865	3.331	0.187
HG-MP-22-1	500,000	67	2.433	2.570	2.876	0.123
HG-MP-22-1	1,000,000	177	2.359	2.463	2.778	0.115
HG-MP-22-1	1,500,000	292	2.320	2.406	2.700	0.105
HG-MP-22-2	100,000	7	5.460	5.961	6.762	0.352
HG-MP-22-2	300,000	33	2.952	3.298	3.512	0.180
HG-MP-22-2	500,000	78	2.767	2.927	3.164	0.118
HG-MP-22-2	1,000,000	200	2.585	2.759	2.881	0.081
HG-MP-22-2	1,500,000	327	2.558	2.709	2.823	0.071
HG-MP-22-3	100,000	6	5.746	6.032	6.365	0.189
HG-MP-22-3	300,000	30	3.479	3.761	4.024	0.170
HG-MP-22-3	500,000	69	3.111	3.348	3.703	0.186
HG-MP-22-3	1,000,000	187	2.852	3.043	3.322	0.150
HG-MP-22-3	1,500,000	313	2.795	2.990	3.214	0.128
HG-MP-22-4	100,000	7	6.886	7.296	8.005	0.314
HG-MP-22-4	300,000	29	3.900	4.310	4.769	0.275
HG-MP-22-4	500,000	64	3.200	3.567	3.772	0.165
HG-MP-22-4	1,000,000	175	2.965	3.211	3.428	0.136
HG-MP-22-4	1,500,000	293	2.934	3.105	3.347	0.110
HG-MP-22-5	100,000	7	5.056	5.460	5.878	0.262
HG-MP-22-5	300,000	33	3.062	3.358	3.662	0.202
HG-MP-22-5	500,000	74	2.788	3.003	3.200	0.130
HG-MP-22-5	1,000,000	197	2.607	2.811	3.003	0.130
HG-MP-22-5	1,500,000	327	2.595	2.777	2.940	0.116

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-23-1	100,000	7	4.849	5.032	5.176	0.109
HG-MP-23-1	300,000	32	3.097	3.298	3.583	0.161
HG-MP-23-1	500,000	68	2.817	2.976	3.283	0.134
HG-MP-23-1	1,000,000	175	2.698	2.852	3.083	0.125
HG-MP-23-1	1,500,000	286	2.680	2.819	3.052	0.123
HG-MP-23-2	100,000	6	5.228	5.433	5.812	0.172
HG-MP-23-2	300,000	31	3.259	3.413	3.694	0.145
HG-MP-23-2	500,000	68	2.822	3.093	3.386	0.153
HG-MP-23-2	1,000,000	171	2.599	2.724	2.935	0.106
HG-MP-23-2	1,500,000	281	2.546	2.678	2.918	0.102
HG-MP-23-3	100,000	6	5.641	6.113	6.425	0.243
HG-MP-23-3	300,000	29	3.180	3.592	3.844	0.205
HG-MP-23-3	500,000	65	2.795	3.147	3.617	0.297
HG-MP-23-3	1,000,000	170	2.517	2.688	2.912	0.134
HG-MP-23-3	1,500,000	285	2.466	2.616	2.868	0.138
HG-MP-23-4	100,000	6	6.290	6.716	7.035	0.251
HG-MP-23-4	300,000	31	3.817	4.090	4.547	0.267
HG-MP-23-4	500,000	70	2.920	3.375	3.832	0.259
HG-MP-23-4	1,000,000	188	2.687	3.171	3.534	0.249
HG-MP-23-4	1,500,000	310	2.676	3.120	3.482	0.245
HG-MP-23-5	100,000	6	5.280	5.587	5.935	0.179
HG-MP-23-5	300,000	31	3.160	3.399	3.676	0.126
HG-MP-23-5	500,000	74	2.984	3.134	3.255	0.097
HG-MP-23-5	1,000,000	192	2.785	2.981	3.233	0.116
HG-MP-23-5	1,500,000	318	2.758	2.938	3.103	0.094
HG-MP-24-1	100,000	6	5.318	5.584	5.862	0.154
HG-MP-24-1	300,000	29	2.858	3.072	3.472	0.161
HG-MP-24-1	500,000	66	2.485	2.783	2.990	0.130
HG-MP-24-1	1,000,000	170	2.435	2.612	2.717	0.103
HG-MP-24-1	1,500,000	279	2.424	2.556	2.677	0.087
HG-MP-24-2	100,000	6	4.789	4.924	5.110	0.113
HG-MP-24-2	300,000	30	2.877	3.070	3.257	0.100
HG-MP-24-2	500,000	70	2.758	2.856	2.985	0.066
HG-MP-24-2	1,000,000	177	2.661	2.741	2.834	0.053
HG-MP-24-2	1,500,000	286	2.597	2.698	2.789	0.064
HG-MP-24-3	100,000	6	4.805	4.967	5.251	0.121
HG-MP-24-3	300,000	30	2.818	2.950	3.215	0.117
HG-MP-24-3	500,000	71	2.560	2.672	2.947	0.108
HG-MP-24-3	1,000,000	186	2.322	2.533	2.641	0.100
HG-MP-24-3	1,500,000	306	2.300	2.507	2.628	0.098
HG-MP-24-4	100,000	5	5.789	6.111	6.468	0.196
HG-MP-24-4	300,000	27	2.739	3.016	3.278	0.160
HG-MP-24-4	500,000	68	2.415	2.721	2.973	0.166
HG-MP-24-4	1,000,000	182	2.373	2.545	2.673	0.093
HG-MP-24-4	1,500,000	301	2.350	2.495	2.639	0.085
HG-MP-24-5	100,000	6	5.640	6.032	6.278	0.222
HG-MP-24-5	300,000	28	3.403	3.593	3.907	0.156
HG-MP-24-5	500,000	70	3.026	3.249	3.510	0.157
HG-MP-24-5	1,000,000	195	2.896	3.052	3.297	0.130
HG-MP-24-5	1,500,000	325	2.839	3.002	3.239	0.120

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-25-1	100,000	5	8.848	9.325	9.966	0.323
HG-MP-25-1	300,000	23	4.380	4.719	5.144	0.251
HG-MP-25-1	500,000	58	3.720	4.153	4.517	0.224
HG-MP-25-1	1,000,000	162	3.601	3.825	3.948	0.114
HG-MP-25-1	1,500,000	272	3.553	3.746	3.918	0.113
HG-MP-25-2	100,000	5	7.750	8.258	8.815	0.347
HG-MP-25-2	300,000	22	4.698	4.991	5.284	0.198
HG-MP-25-2	500,000	53	4.079	4.441	4.669	0.186
HG-MP-25-2	1,000,000	147	3.806	3.998	4.204	0.106
HG-MP-25-2	1,500,000	250	3.721	3.919	4.092	0.099
HG-MP-25-3	100,000	5	9.253	10.002	10.738	0.410
HG-MP-25-3	300,000	23	4.541	5.301	5.869	0.330
HG-MP-25-3	500,000	53	3.902	4.300	4.625	0.216
HG-MP-25-3	1,000,000	144	3.548	3.814	4.157	0.196
HG-MP-25-3	1,500,000	240	3.540	3.746	4.031	0.187
HG-MP-25-4	100,000	6	10.317	10.770	11.175	0.250
HG-MP-25-4	300,000	26	6.737	7.139	7.647	0.296
HG-MP-25-4	500,000	57	5.489	6.035	6.360	0.258
HG-MP-25-4	1,000,000	157	5.038	5.307	5.531	0.162
HG-MP-25-4	1,500,000	267	4.911	5.171	5.340	0.153
HG-MP-25-5	100,000	6	7.093	7.418	7.809	0.241
HG-MP-25-5	300,000	26	4.073	4.571	4.953	0.271
HG-MP-25-5	500,000	57	3.687	3.962	4.311	0.168
HG-MP-25-5	1,000,000	154	3.490	3.623	3.705	0.063
HG-MP-25-5	1,500,000	257	3.486	3.584	3.662	0.052
HG-MP-26-1	100,000	5	8.500	8.946	9.759	0.346
HG-MP-26-1	300,000	23	4.648	4.971	5.200	0.158
HG-MP-26-1	500,000	55	4.033	4.301	4.724	0.185
HG-MP-26-1	1,000,000	148	3.822	3.996	4.229	0.119
HG-MP-26-1	1,500,000	244	3.801	3.947	4.128	0.096
HG-MP-26-2	100,000	5	9.118	9.329	9.730	0.194
HG-MP-26-2	300,000	25	4.657	5.186	5.594	0.287
HG-MP-26-2	500,000	57	4.350	4.713	5.083	0.250
HG-MP-26-2	1,000,000	150	4.088	4.255	4.495	0.133
HG-MP-26-2	1,500,000	249	3.945	4.140	4.384	0.138
HG-MP-26-3	100,000	5	8.255	8.520	8.806	0.193
HG-MP-26-3	300,000	25	4.617	5.086	5.339	0.204
HG-MP-26-3	500,000	58	4.115	4.450	4.747	0.207
HG-MP-26-3	1,000,000	158	3.864	4.158	4.395	0.154
HG-MP-26-3	1,500,000	263	3.785	4.074	4.278	0.153
HG-MP-26-4	100,000	6	7.665	8.035	8.558	0.322
HG-MP-26-4	300,000	26	4.660	5.010	5.257	0.170
HG-MP-26-4	500,000	60	3.775	4.069	4.428	0.232
HG-MP-26-4	1,000,000	163	3.370	3.644	4.081	0.224
HG-MP-26-4	1,500,000	268	3.336	3.566	4.069	0.197
HG-MP-26-5	100,000	5	10.069	10.737	12.767	0.775
HG-MP-26-5	300,000	21	4.860	5.238	5.915	0.269
HG-MP-26-5	500,000	50	4.257	4.488	4.959	0.201
HG-MP-26-5	1,000,000	143	3.660	3.977	4.101	0.119
HG-MP-26-5	1,500,000	243	3.623	3.916	3.996	0.105

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-27-1	100,000	5	8.241	8.685	9.215	0.337
HG-MP-27-1	300,000	24	3.929	4.726	5.600	0.457
HG-MP-27-1	500,000	58	3.470	3.913	4.231	0.216
HG-MP-27-1	1,000,000	153	3.240	3.647	4.113	0.253
HG-MP-27-1	1,500,000	249	3.207	3.565	4.004	0.253
HG-MP-27-2	100,000	5	8.303	8.962	9.490	0.339
HG-MP-27-2	300,000	25	4.541	5.029	5.419	0.269
HG-MP-27-2	500,000	59	3.987	4.495	4.944	0.308
HG-MP-27-2	1,000,000	154	3.917	4.284	4.643	0.253
HG-MP-27-2	1,500,000	251	3.872	4.189	4.570	0.239
HG-MP-27-3	100,000	4	11.815	13.338	14.740	0.994
HG-MP-27-3	300,000	18	3.595	4.198	4.941	0.446
HG-MP-27-3	500,000	50	3.209	3.492	3.948	0.242
HG-MP-27-3	1,000,000	143	3.127	3.307	3.690	0.179
HG-MP-27-3	1,500,000	238	3.105	3.279	3.666	0.179
HG-MP-27-4	100,000	5	10.554	11.139	12.130	0.483
HG-MP-27-4	300,000	21	5.667	6.019	6.377	0.209
HG-MP-27-4	500,000	50	4.849	5.259	5.543	0.213
HG-MP-27-4	1,000,000	139	4.635	4.878	5.097	0.132
HG-MP-27-4	1,500,000	231	4.586	4.779	4.931	0.118
HG-MP-27-5	100,000	5	7.028	7.555	8.051	0.276
HG-MP-27-5	300,000	25	4.292	4.711	4.983	0.187
HG-MP-27-5	500,000	57	4.046	4.255	4.500	0.127
HG-MP-27-5	1,000,000	151	3.750	4.021	4.166	0.117
HG-MP-27-5	1,500,000	249	3.687	3.982	4.136	0.122
HG-MP-28-1	100,000	5	14.302	15.472	16.244	0.586
HG-MP-28-1	300,000	21	7.162	7.688	8.135	0.322
HG-MP-28-1	500,000	47	6.129	6.440	7.299	0.336
HG-MP-28-1	1,000,000	135	4.861	5.392	5.943	0.299
HG-MP-28-1	1,500,000	234	4.825	5.281	5.700	0.265
HG-MP-28-2	100,000	5	9.033	9.488	10.233	0.385
HG-MP-28-2	300,000	22	5.498	5.831	6.135	0.208
HG-MP-28-2	500,000	50	4.772	4.959	5.243	0.137
HG-MP-28-2	1,000,000	135	4.170	4.428	4.722	0.167
HG-MP-28-2	1,500,000	223	4.104	4.285	4.580	0.126
HG-MP-28-3	100,000	5	11.762	12.479	13.480	0.559
HG-MP-28-3	300,000	20	6.084	6.585	7.036	0.313
HG-MP-28-3	500,000	48	5.228	5.461	5.791	0.145
HG-MP-28-3	1,000,000	134	4.731	4.974	5.302	0.143
HG-MP-28-3	1,500,000	224	4.655	4.893	5.152	0.121
HG-MP-28-4	100,000	5	13.457	14.776	15.516	0.633
HG-MP-28-4	300,000	21	7.402	7.974	8.524	0.355
HG-MP-28-4	500,000	46	6.330	6.716	7.224	0.270
HG-MP-28-4	1,000,000	127	5.502	5.815	6.458	0.242
HG-MP-28-4	1,500,000	215	5.389	5.639	6.011	0.177
HG-MP-28-5	100,000	5	11.717	12.876	13.534	0.547
HG-MP-28-5	300,000	20	6.120	6.588	6.938	0.195
HG-MP-28-5	500,000	47	5.200	5.612	5.856	0.186
HG-MP-28-5	1,000,000	132	4.948	5.189	5.515	0.168
HG-MP-28-5	1,500,000	225	4.838	5.078	5.427	0.175

Network	# Function Evaluations	Mean # Iterations	Min Cost (in 10^6 EUR)	Mean Cost (in 10^6 EUR)	Max Cost (in 10^6 EUR)	Stddev Cost (in 10^6 EUR)
HG-MP-29-1	100,000	4	14.410	14.959	15.641	0.391
HG-MP-29-1	300,000	17	6.257	6.851	7.230	0.274
HG-MP-29-1	500,000	41	5.417	5.735	6.186	0.244
HG-MP-29-1	1,000,000	117	4.778	5.062	5.592	0.247
HG-MP-29-1	1,500,000	198	4.573	4.884	5.499	0.240
HG-MP-29-2	100,000	5	16.444	17.201	17.823	0.481
HG-MP-29-2	300,000	20	8.337	9.329	11.109	0.855
HG-MP-29-2	500,000	44	6.069	7.131	8.155	0.664
HG-MP-29-2	1,000,000	122	5.017	5.703	6.354	0.431
HG-MP-29-2	1,500,000	207	4.849	5.560	6.256	0.433
HG-MP-29-3	100,000	5	14.855	15.869	16.619	0.623
HG-MP-29-3	300,000	19	7.681	8.655	9.365	0.575
HG-MP-29-3	500,000	43	5.592	6.125	6.899	0.345
HG-MP-29-3	1,000,000	128	5.057	5.268	5.750	0.208
HG-MP-29-3	1,500,000	220	4.952	5.142	5.552	0.180
HG-MP-29-4	100,000	5	10.829	11.970	12.557	0.583
HG-MP-29-4	300,000	20	5.808	6.507	7.228	0.437
HG-MP-29-4	500,000	47	4.842	5.515	6.538	0.500
HG-MP-29-4	1,000,000	130	4.634	4.965	5.531	0.306
HG-MP-29-4	1,500,000	215	4.617	4.789	5.139	0.164
HG-MP-29-5	100,000	5	12.781	13.652	14.963	0.558
HG-MP-29-5	300,000	21	6.976	7.236	7.459	0.174
HG-MP-29-5	500,000	50	5.965	6.242	6.573	0.219
HG-MP-29-5	1,000,000	137	5.492	5.740	6.058	0.170
HG-MP-29-5	1,500,000	230	5.438	5.619	5.932	0.150
HG-MP-30-1	100,000	5	12.217	13.480	14.729	0.911
HG-MP-30-1	300,000	19	6.766	7.087	7.380	0.184
HG-MP-30-1	500,000	44	5.801	6.267	6.518	0.221
HG-MP-30-1	1,000,000	120	5.097	5.470	5.839	0.222
HG-MP-30-1	1,500,000	204	4.878	5.298	5.759	0.235
HG-MP-30-2	100,000	4	13.930	15.653	16.982	0.971
HG-MP-30-2	300,000	17	6.030	6.555	7.026	0.278
HG-MP-30-2	500,000	43	4.803	5.357	5.764	0.287
HG-MP-30-2	1,000,000	118	4.242	4.749	5.238	0.293
HG-MP-30-2	1,500,000	195	4.142	4.585	5.019	0.271
HG-MP-30-3	100,000	5	11.964	12.701	13.300	0.358
HG-MP-30-3	300,000	20	7.102	7.397	7.777	0.240
HG-MP-30-3	500,000	46	5.744	6.096	6.395	0.226
HG-MP-30-3	1,000,000	127	4.891	5.412	6.154	0.318
HG-MP-30-3	1,500,000	214	4.820	5.301	5.885	0.278
HG-MP-30-4	100,000	5	9.219	9.739	10.796	0.445
HG-MP-30-4	300,000	20	5.297	5.783	6.497	0.317
HG-MP-30-4	500,000	49	4.966	5.413	5.974	0.327
HG-MP-30-4	1,000,000	128	4.530	4.847	5.207	0.260
HG-MP-30-4	1,500,000	215	4.432	4.629	5.021	0.169
HG-MP-30-5	100,000	4	12.417	13.202	13.632	0.362
HG-MP-30-5	300,000	17	5.356	5.751	6.004	0.220
HG-MP-30-5	500,000	43	4.525	4.804	5.023	0.175
HG-MP-30-5	1,000,000	123	4.096	4.425	4.759	0.214
HG-MP-30-5	1,500,000	205	4.051	4.323	4.681	0.205

References

1. World Health Organization. *Guidelines for Drinking-Water Quality*, 4th ed.; World Health Organization (WHO): Geneva, Switzerland, 2011.
2. Mays, L.; Koutsoyiannis, D.; Angelakis, A. A brief history of urban water supply in antiquity. *Water Supply* **2007**, *7*, 1–12.
3. United Nations Children's Emergency Fund; World Health Organization (WHO). *Progress on Drinking Water and Sanitation*; WHO: Geneva, Switzerland, 2012.

4. Grayman, W.; Loucks, D.; Saito, L. *Towards a Sustainable Water Future: Visions for 2050*; American Society of Civil Engineers (ASCE): Reston, VA, USA, 2012.
5. Alperovits, A.; Shamir, U. Design of optimal water distribution systems. *Water Resour. Res.* **1977**, *13*, 885–900.
6. Quindry, G.; Liebman, J.; Brill, E. Optimization of looped water distribution systems. *J. Environ. Eng. Div.* **1981**, *107*, 665–679.
7. Kessler, A.; Shamir, U. Analysis of the linear programming gradient method for optimal design of water supply networks. *Water Resour. Res.* **1989**, *25*, 1469–1480.
8. Shamir, U. Optimal design and operation of water distribution systems. *Water Resour. Res.* **1974**, *10*, 27–36.
9. El-Bahrawy, A.; Smith, A. A methodology for optimal design of pipe distribution networks. *Can. J. Civ. Eng.* **1987**, *14*, 207–215.
10. Fujiwara, O.; Khang, D. A two-phase decompostion method for optimal design of looped water distribution networks. *Water Resour. Res.* **1990**, *26*, 539–549.
11. Duan, N.; Mays, L.; Lansey, K. Optimal reliability-based design of pumping and distribution systems. *J. Hydraul. Eng.* **1990**, *116*, 249–268.
12. Rossman, L. *EPANET 2: Users Manual*; Environmental Protection Agency: Washington, DC, USA, 2000.
13. Loganathan, G.; Greene, J.; Ahn, T. Design heuristic for globally minimum cost water distribution systems. *J. Water Resour. Plan. Manag.* **1995**, *121*, 182–192.
14. Cunha, M.; Sousa, J. Hydraulic infrastructures design using simulated annealing. *J. Infrastruct. Syst.* **2001**, *7*, 32–39.
15. Cunha, M.; Ribeiro, L. Tabu search algorithms for water network optimization. *Eur. J. Oper. Res.* **2004**, *157*, 746–758.
16. Maier, H.; Simpson, A.; Zecchin, A.; Foong, W.; Phang, K.; Seah, H.; Tan, C. Ant colony optimization for design of water distribution systems. *J. Water Resour. Plan. Manag.* **2003**, *129*, 200–209.
17. Zecchin, A.; Simpson, A.; Maier, H.; Nixon, J. Parametric study for an ant algorithm applied to water distribution system optimization. *IEEE Trans. Evol. Comput.* **2005**, *9*, 175–191.
18. Dandy, G.; Simpson, A.; Murphy, L. An improved genetic algorithm for pipe network optimisation. *Water Resour. Res.* **1996**, *32*, 449–458.
19. Gupta, I.; Gupta, A.; Khanna, P. Genetic algorithm for optimization of water distribution systems. *Environ. Model. Softw.* **1999**, *14*, 437–446.
20. Bi, W.; Dandy, G.; Maier, R. Improved genetic algorithm optimization of water distribution system design by incorporating domain knowledge. *Environ. Model. Softw.* **2015**, *69*, 370–381.
21. Lin, M.; Liu, Y.; Liu, G.; Chu, C. Scatter search heuristic for least-cost design of water distribution networks. *Eng. Optim.* **2007**, *39*, 857–876.
22. Vasan, A.; Simonovic, S. Optimization of water distribution network design using differential evolution. *J. Water Resour. Plan. Manag.* **2010**, *136*, 279–287.
23. Marchi, A.; Dandy, G.; Wilkins, A.; Rohrlach, H. Methodology for comparing evolutionary algorithms for optimization of water distribution systems. *J. Water Resour. Plan. Manag.* **2014**, *140*, 22–31.
24. De Corte, A.; Sörensen, K. Optimisation of gravity-fed water distribution network design: A critical review. *Eur. J. Oper. Res.* **2013**, *228*, 1–10.
25. Farmani, R.; Walters, G.; Savic, D. Trade-off between total cost and reliability for Anytown water distribution network. *J. Water Resour. Plan. Manag.* **2005**, *131*, 161–171.
26. Bragalli, C.; D'Ambrosio, C.; Lee, J.; Lodi, A.; Toth, P. On the optimal design of water distribution networks: A practical MINLP approach. *Optim. Eng.* **2012**, *13*, 219–246.
27. De Corte, A.; Sörensen, K. HydroGen: An artificial water distribution network generator. *Water Resour. Manag.* **2014**, *28*, 333–350.
28. Yates, D.; Templeman, A.; Boffey, T. The computational complexity of the problem of determining least capital cost designs for water supply networks. *Eng. Optim.* **1984**, *7*, 143–155.
29. Lourenço, H.; Martin, O.; Stützle, T. Chapter 11: Iterated local search. In *Handbook of Metaheuristics*; Springer: New York, NY, USA, 2003; pp. 320–353.
30. López-Ibáñez, M. Operational Optimisation of Water Distribution Networks. Ph.D. Thesis, School of Engineering and the Built Environment, Edinburgh Napier University (UK), Edinburgh, UK, 2009.

31. López-Ibáñez, M. Extended EPANET Toolkit. Available online: http://iridia.ulb.ac.be/ manuel/epanetlinux. html (accessed on 18 August 2016).
32. Maier, H.; Kapelan, Z.; Kasprzyk, J.; Kollat, J.; Matott, L.; Cunha, M.; Dandy, G.; Gibbs, M.; Keedwell, E.; Marchi, A.; et al. Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions. *Environ. Model. Softw.* **2014**, *62*, 271–299.
33. De Corte, A.; Sörensen, K. Hydrogen. Available online: http://antor.uantwerpen.be/hydrogen (accessed on 18 August 2016).