Optimizing access to drinking water in remote areas. Application to Nepal

Gilbert Laporte

HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7
School of Management, University of Bath, Claverton Down, Bath BA2 7AY, United Kingdom

Marie-Ève Rancourt

HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7

Jessica Rodríguez-Pereira[1]

HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7
Universitat Pompeu Fabra, Ramon Trias Fargas, 25-27 08005 Barcelona, Spain

Selene Silvestri

HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7
FICO, 3 Piazza Filippo Meda 20121 Milan, Italy

Abstract

This study is motivated by the need to restore part of the Nepal water distribution network that was destroyed by the Gorkha and Dolakha earthquakes in April and May 2015. The problem consists of two hierarchical subproblems: locating water taps to ensure a good coverage of the population, and connecting these water taps to water sources by means of a pipe distribution network. Both subproblems are subject to a variety of accessibility and technical constraints that make the problem unique and highly complex. Namely, because Nepal is highly mountainous, elevations must be taken into account in the distance calculations, and the distribution network is gravity-fed, meaning that pumps are not used. The problem is solved by means of a two-phase matheuristic: the first subproblem is a constrained location-allocation problem which is solved exactly by integer linear programming, while the second subproblem is tackled by means of a cluster-first, tree-second heuristic. Several variants of the heuristic are developed and compared. The network design problem is of very large scale, being solved on a graph with as many as 29,900 vertices and 75,200 arcs. Tests are performed on real-world data, obtained by satellite imagery, from the Suspa Kshemawati and Lapilang communities in the Dolakha district. Extensive computational results confirm the effectiveness of the proposed methodology

[1]Corresponding author: jessica.rodriguez@upf.edu

and enable an identification of the best parameter settings and algorithmic tactical choices.

Keywords: Water distribution network design, humanitarian logistics, location-allocation, Steiner forest, gravity-fed system, Dolakha district

1. Introduction

The lack of access to drinking water is one of the most important humanitarian and development problems in remote areas of developing countries. The efforts made to improve the distribution of clean water, and to provide better sanitation and hygiene help protect vulnerable populations from waterborne diseases, such as intestinal worms, typhoid fever, and cholera, which is one of the leading causes of death in the world. A report from WHO[2] and UNICEF (2019) states that in 2017 the poor and rural communities were the most at risk and it was estimated that only 71 percent of the world population could benefit from a safely managed drinking water service, i.e., drinking water from an improved water source that is located on premises, available when needed, free from faecal and priority chemical contamination, and within a collection time that is no more than 30 minutes for a round trip, including queueing. Ensuring global access to clean water figures among the 17 Sustainable Development Goals (SDG) established by the United Nations General Assembly (2015) in the 2030 Agenda for Sustainable Development. In particular, the SDG 6, called "Clean water and sanitation", seeks to achieve universal and equitable access to safe and affordable drinking water for all. For a developing country such as Nepal, which is the focus of this study, it is challenging to achieve this goal given its poor infrastructure and high population growth (Udmale et al., 2016).

Improving the resilience of water, sanitation and hygiene systems is key to the security issue of sustainable development of Nepal (Shrestha and Dahal, 2020), as failure to make water services resilient to disasters can have negative impacts on human health and well-being, as well as on the economy, productivity, and development (Aihara et al., 2018). This goal is promoted by WASH United, a non-profit organization that operates in several countries. The present study is motivated by the case of the two earthquakes that hit the Gorkha and Dolakha districts of Nepal on 25 April and 12 May, 2015 (see Figure 1), and greatly impaired the provision of health-care and water, sanitation and hygiene services (Bagcchi, 2015). These earthquakes destroyed over half a million houses, and severely damaged services and infrastructures, making an innumerable number of Nepalis homeless (National Planning Commission, Kathmandu, Nepal 2015; Nepal

[2]Appendix A contains a glossary of the abbreviations used in this paper.

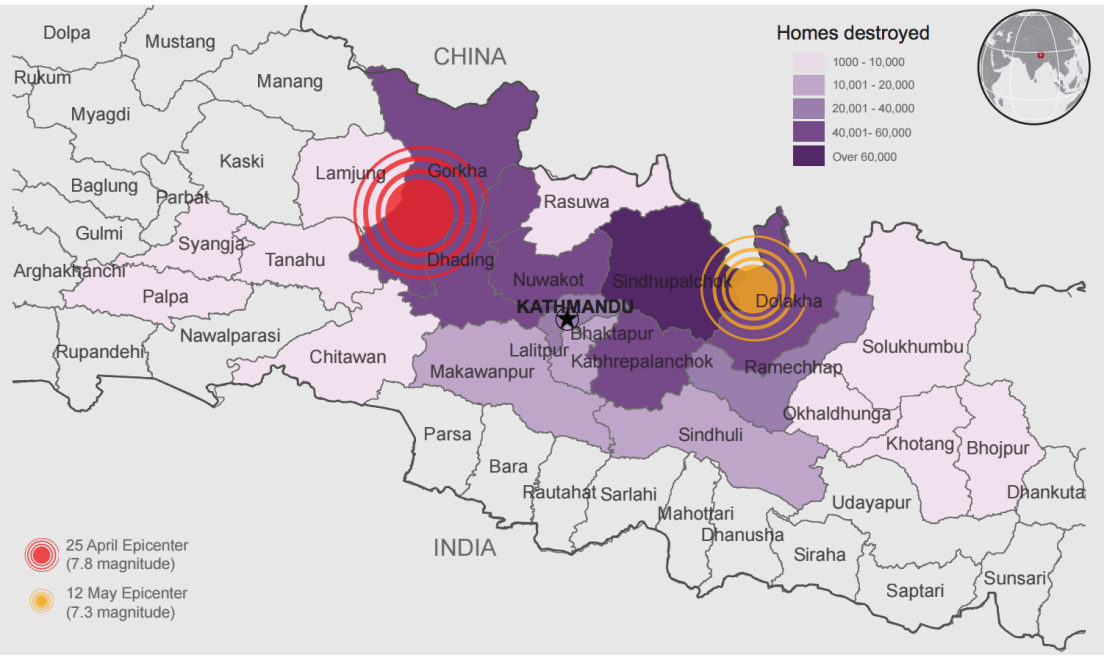

Figure 1: Epicentres of the Gorkha and Dolakha earthquakes. Source: Goldberg (2015).

Red Cross Society, 2017). Prior to the earthquakes, it was already difficult for the Government of Nepal to fulfil potable water supply needs, and the damages of the existing infrastructure have made this task even more challenging (Mishra and Acharya, 2018). Many of the most affected areas were rural, and some of them were difficult to reach. More than 1.1 million people were left without access to a protected water source, as the water supply systems across 14 districts were destroyed (Office for the Coordination of Humanitarian Affairs, 2015). After an initial emergency response phase, an Earthquake Response Operation was established in the affected districts. Part of its program embraced the rehabilitation of 18 damaged community water supply systems in the Dolakha district (Figure 2), which is our region of interest (RoI), where the main sources to drinking water were gravity-fed systems with community tap stands (Riedler et al., 2017).

In this context, with the collaboration of the Interfaculty Department of Geoinformatics (Z-GIS) of the University of Salzburg, and of the Austrian Red Cross Society (AutRC), which cooperated with the National Red Cross Societies of Nepal and Switzerland, a proposal was made to develop an optimization tool to support the rehabilitation of the community water supply systems and provide a database for future situations. Data collection during a disaster poses

many challenges, including secure access to the affected area, preparing resources for the data collection, obtaining informed consent from the affected population, fragmenting data collection, and reporting among different relief teams (Kubo et al., 2019). Because the lack of adequate and accurate data may hinder the rehabilitation work, it is crucial to gather and extract the relevant information from raw data from all available sources into a structured dataset that can be later exploited to tackle the rehabilitation and design of water supply systems. As noted by Nomura et al. (2021), the use of geographic information systems for operational research can help develop location analysis and obtain more accurate data on spatial elements, such as road networks and geographic obstacles.

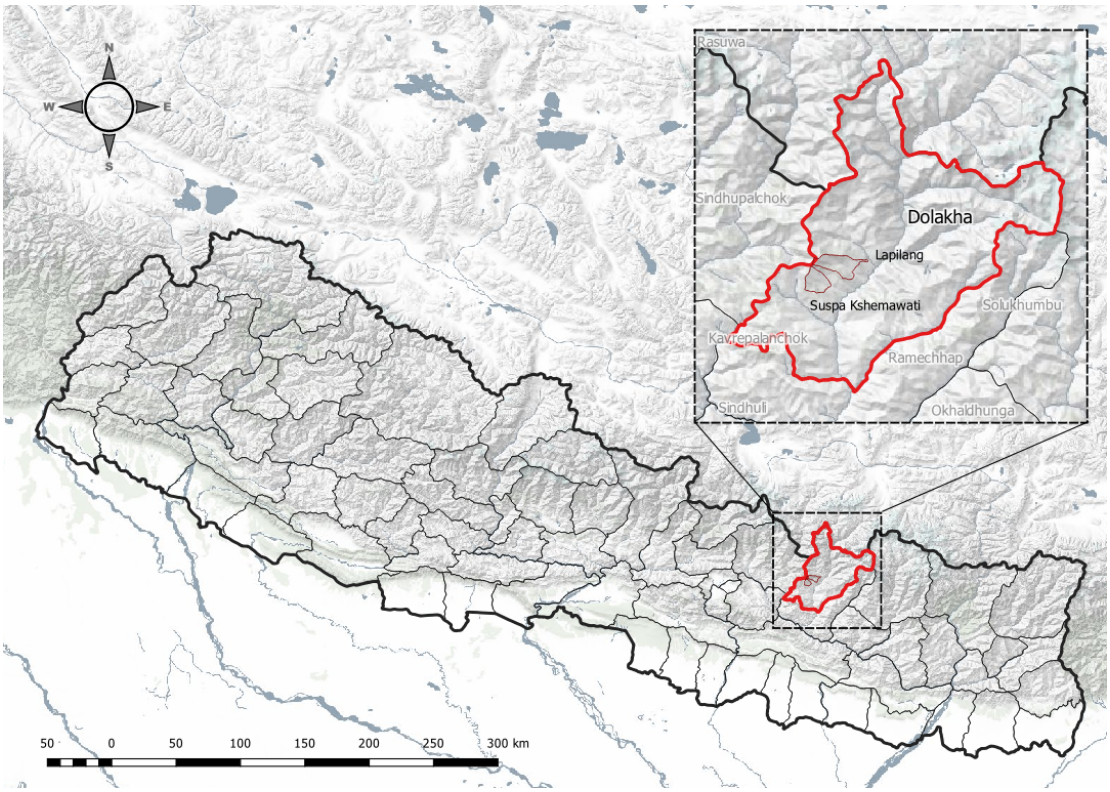

Figure 2: Topographic map of Nepal showing the Dolakha district.[3]

[3]This map and all subsequent ones were generated with QGIS 3.18 in the WGS84 reference system and represented in the UTM (Zone 45, Northern hemisphere) projection, with coordinates expressed in meters. These maps were produced by Joel Grau Bellet, a professional cartographer, and the authors.

1.1. Problem definition

The goal of this work is to model and solve a Water Supply System Design Problem (WSSDP) based on a gravity-fed system (see Section 2) which consists of two subproblems. The first is called the Water Tap Location-Allocation Problem (WTLAP) whose aim is to determine the number and the locations of community water taps (WTs) and the assignment of households to WTs under certain accessibility standards. The second subproblem is called the Water Distribution Network Design Problem (WDNDP) whose aim is to identify a low-cost network connection from the water sources (WSs) to the WTs identified in the first subproblem, while respecting the technical specifications of a gravity-fed water system. In this paper we develop a methodology capable of solving the WSSDP from scratch, but in practice this may not be necessary since the planners may wish to make use of parts of the infrastructure that are still functional. After an earthquake, determining which parts of the network can be reused falls under the responsibility of civil engineers. In our application we were told that it would be preferable to design a full network. However, our methodology can easily be adapted to the case of a partial network reconstruction, as we explain at the beginning of Section 3.

The primary objective of the humanitarian organizations handling the aftermath of the earthquakes is to maximize the population's access to a safe water point respecting the WASH standars, which translates into solving the WTLAP. According to National Red Cross Societies of Nepal, Austria and Switzerland involved in the rehabilitation program of our RoI, connecting the WTs identified by the WTLAP is only a secondary objective since the cost of installing water pipes is not nearly as important as ensuring the health and welfare of the population. For this reason, the WTLAP and the WDNDP must be solved hierarchically, which also has the advantage of simplifying the solution process.

Defined on a directed graph, a feasible solution for the WSSDP is a Steiner forest, i.e., a collection of Steiner trees[4] (Gilbert and Pollak, 1968), each rooted in a WS which together span all WTs and assign households to WTs. Like a minimum spanning tree (MST), a Steiner tree spans all vertices of a graph, but there is a key difference between the two problems. In an MST, only the links that directly connect vertices of the graph may be used, whereas in a Steiner tree extra vertices, called Steiner vertices, may be added to the graph in order to reduce the cost of the solution. From a computational point of view the MST is an easy problem that can be

[4]Strictly speaking, these are arborescences, i.e., directed trees in which every vertex can be reached from a root vertex, here a WS (Gross et al., 2013). However, for simplicity of writing, and since no confusion can arise, we use the word tree.

solved in polynomial time by means of a greedy algorithm (Prim, 1957), whereas the Steiner tree problem is NP-hard and very difficult to solve in practice. Figure 3 depicts two feasible solutions for the WSSDP, with four trees if there is no Steiner vertex, or three trees if a Steiner vertex is used. On the left, each of the two WSs serves two trees from different access points, while on the right two of the trees are merged into one through the Steiner vertex, resulting in a shorter network.

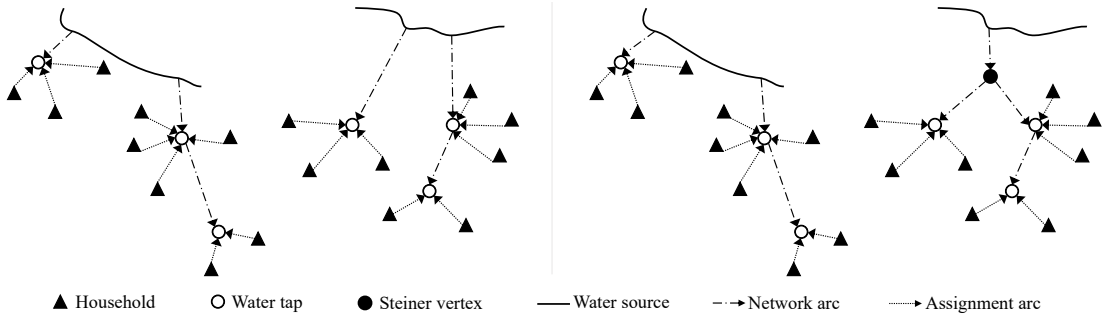

Figure 3: Examples of solutions for the WSSDP without and with a Steiner vertex.

1.2. Positioning of this study within several streams of literature

The WSSDP belongs to the class of General Network Design Problems (Contreras and Fernández, 2012) which combines two important research areas in network optimization: location analysis and network design. In our context these two components are the WTLAP and the WDNDP. The WSSDP is related to the Connected Facility Location Problem (ConFLP) which aims to identify a minimum cost assignment of each user to exactly one opened facility and to interconnect the opened facilities via a Steiner tree. However, there are several important structural differences between the WSSDP treated in this paper and the classical ConFLP: 1) any feasible solution of the WSSDP has to respect some technical requirements of a water supply system based on a gravity flow, a constraint not considered in the ConFLP; 2) unlike the ConFLP, a feasible WSSDP solution is represented by a Steiner forest, where some WSs are root vertices of the Steiner trees, but not all of them are necessarily used; 3) the WSSDP considers capacity constraints for the WSs and the WTs; and 4) in the WSSDP the households assignments have to respect specific foot travel restrictions (see Section 2.1).

Several approaches have been proposed to solve network design problems arising in the contexts of disaster relief and development programs, in which location decisions are an important component (see the reviews of Dönmez et al. (2021) and Trivedi and Singh (2018)). However,

to the best of our knowledge only one article describes the use of spanning trees in this context (Karsu et al., 2021). Regarding the literature on locational decisions in humanitarian logistics, there are two main streams: the location of warehouses and that of service points. International and national warehouse location decisions are often considered for prepositioning equipment and relief items, (e.g., Balcik and Beamon, 2008; Jahre et al., 2016; Acimovic and Goentzel, 2016; Arnette and Zobel, 2019). Maharjan and Hanaoka (2017) determine an optimal set of warehouses for humanitarian relief by solving a covering location problem adapted to the case of Nepal. We refer the reader to Balcik et al. (2016), Kara and Rancourt (2019) and Sabbaghtorkan et al. (2020) for reviews of prepositioning problems. Here we aim to locate WTs in rural and remote areas, which could be considered as distribution or dispensing points in a humanitarian setting, i.e., locations where beneficiaries are served. In line with this objective, Rancourt et al. (2015) determined a set of food aid distribution points in Kenya by considering the joint welfare of the humanitarian agencies and of the beneficiaries. As in our study, spatial data were processed to determine the parameters of the last-mile distribution network, optimized by solving classical location models, where the welfare of the beneficiaries was modeled as a function of their walking distance. VonAchen et al. (2016) and Cherkesly et al. (2019) solved a location-routing covering problem arising in a community healthcare program, whose aim was to cover underserved areas of Liberia. Other applications combining coverage and routing problems for relief distribution and healthcare delivery considered walking distances. For example, Nolz et al. (2010) and Naji-Azimi et al. (2012) adapted classical coverage-routing problems to optimize relief distribution in disaster-affected areas, while Doerner et al. (2007) and Hodgson et al. (1998) adapted them to locate mobile clinics in developing countries. Tatham et al. (2010) presented a bi-objective location-routing model to plan water distribution tours by considering features such as transportation modes, road categories, and a coverage constraint. However, these problems aim to plan service delivery tours as opposed to designing a distribution network, as we do. Moreover, in our case, the location decisions related to WTs are prioritized as a primary objective, which makes the location and network design problems separable.

In parallel, there exists a rich literature on the urban water network design problem, where physical features such as the length and diameter of the pipelines and a coverage radius must be determined (see D'Ambrosio et al. (2015) for a survey on drinking water distribution network optimization problems). In these studies, the methodologies are mostly designed for cities where the locations of both the demand nodes and pipes are already known, whereas in our case these decisions have to be made. Karsu et al. (2021) studied a water network design problem quite

similar to ours, to distribute water in refugee camps. They proposed a bi-objective integer programming model to determine the locations of WTs and the design of the network. Their first objective was to minimize the total length of the network, while the second objective was to maximize accessibility by reducing the walking distance to the WTs. Their problem was solved by means of exact and heuristic algorithms to identify Pareto-optimal solutions. Unlike these authors, we consider complex technical constraints brought by the need to consider of a gravity-fed system, and there is no limitations on the number of WTs to open. The network topology proposed in Karsu et al. (2021) is a tree that does not use Steiner vertices to help reduce the length of the network. Moreover, while their instances are also based on real data, they contain only 38 or 73 vertices, whereas our instances are larger by three orders of magnitude.

1.3. Scientific contributions and organization of this paper

Our main scientific contribution is to introduce, model and solve a complex network design problem arising in the distribution of drinking water in remote mountainous areas, with an emphasis on Nepal. The problem is new and highly relevant in the context of humanitarian development projects. It is also difficult to solve because of its continuous nature and its very large scale. For example, in our test instances, the underlying graph contains as many as 29,900 vertices and 75,200 arcs. In addition, the need to respect some WASH national accessibility standards and to satisfy some technical constraints associated with the feasibility of a gravity-fed system results in several modeling and algorithmic difficulties. Taken together, these features make our problem very different from those studied in the network design literature, and considerably more difficult to solve than most.

The first critical step in solving the problem lies in data collection which by itself poses several challenges in a humanitarian context. We obtained data from satellite imagery, technical reports and the AutRC. The raw data had to be processed in order to create the graph that constitutes the basis of our mathematical model. While distance computations pose no major challenges in most network design problems, here we had to adapt the classical Dijkstra (1959) shortest path algorithm to account for three different distance types that can occur in the context of the WSSDP, namely some tridimensional distances due to the topography of the RoI.

Because our problem is of very large scale and high complexity, it is impractical to solve it exactly. We have therefore developed an efficient matheuristic that exploits the separable nature of the WSSDP into the WTLAP and the WDNDP. The first subproblem is solved exactly through integer linear programming, while the second subproblem is solved heuristically through

simulated annealing (SA). The SA phase performs a neighbourhood search over a set of potential Steiner vertices and solves GMSTs, exactly for the given selections of Steiner vertices, i.e., MSTs constrained by the technical requirements of a gravity-fed water distribution system. The overall procedure is obviously heuristic since the output of the WTLAP influences the solution of the WDNDP, and the SA heuristic employed for the solution of the WDNDP performs a limited search.

We solved two real instances of the problem on two Village Development Committees (VDCs) data sets in the Dolakha district. We performed extensive numerical analyses to assess the impact of several algorithmic variants, and to generate managerial insights.

The remainder of this paper is organized as follows. Section 2 presents the main features of the WSSDP. Section 3 describes the proposed matheuristic. Section 4 reports the computational results. Conclusions follow in Section 5.

2. The Water Supply System Design Problem

The main sources of drinking water in remote mountainous areas of Nepal are gravity-fed community systems where people have to walk daily to fetch water at community WTs. The use of gravity-fed systems is often advocated for water distribution in developing countries since these systems do not rely on pumps which are costly. They also require a source of energy as well as maintenance, which entail extra expenditures and expertise. Indeed, pump-operated systems often make use of foreign equipment that is unfamiliar to the local populations (Oxfam, 2019; GC et al., 2021; Ambuehl et al., 2021). Moreover, due to the terrain which is usually rough and mountainous in Nepal, people are often required to walk over large distances in a steep landscape to reach a WT. The goal of the WSSDP is to design a community gravity-fed system whose emphasis is the location of the community WTs in such a way that walking distances for the assigned households meet the WASH national standards. Additionally, the network cost incurred to connect WSs to WTs should be minimized. We now describe the WTLAP and the WDNDP which are the two components of the WSSDP.

2.1. The Water Tap Location-Allocation Problem

The WTLAP consists of identifying optimal locations of WTs, and assigning households to these locations in such a way that the users do not have to walk more than a preset distance to reach a WT, based on the WASH standards. An optimal solution of the WTLAP must provide a good trade-off between the minimization of the walking distances and the number of opened

WTs. To limit the walking distances we apply a standard which determines the maximum horizontal and vertical distances between the households and the WTs. In some exceptional cases these standards can be relaxed. We denote by (Δ, Σ) the preferred maximum horizontal and vertical distances, and by $(\bar{\Delta}, \bar{\Sigma})$ the longer walking distances allowed in exceptional cases. Furthermore, two additional constraints are imposed on the solution: 1) a maximum number of households that can be assigned to the same WT, and 2) some points of interest (PoIs) which represent facilities such as hospitals or schools requiring a dedicated WT.

It is worth noting that distances alone do not fully capture the effort expended by users to fetch their water since these do not take into account the weights that are transported. In this context, an alternative to distance is the *work*, measured in joules, which is a function of distance, weight, and angle of displacement (see Young and Freedman, 2020). Thus, a user equidistant from a WT called A (downhill) and a WT called B (uphill) will likely prefer to fetch his water from B since it is easier to bring back a container full of water from B than from A. Using work measures instead of distances in our model and algorithms is straightforward.

2.2. The Water Distribution Network Design Problem

The WDNDP consists of determining a minimum cost water supply network. Once the locations of the WTs have been identified, and the total daily demand of water per WT has been computed based on the assigned households, water has to be channeled from WSs to WTs while ensuring that the capacity of each WS is respected and that the connections from WSs to WTs satisfy some technical requirements for a gravity-fed system: 1) A WS cannot provide water to a WT if the WT location or any part of the pipes connecting them is on a higher elevation than the WS. This is a necessary but not sufficient condition to ensure that a WT can receive water. 2) Let e_{WS} and e_v denote the elevation of a WS and of a generic point v, such that $e_{WS} - e_v \geq 0$, and let $d_{WS,v}$ be the distance in the three-dimensional space from the WS to v, along a path such that none of the points in it has a higher elevation than e_{WS}. Vertex v can receive water from the WS only if

$$e_{WS} - e_v - \gamma d_{WS,v} \geq 0 \tag{1}$$

for some positive coefficient γ. This formula is based on the Hazen-Williams equation (Williams and Hazen, 1933), which is an empirical relationship relating the flow of water in a pipe with its physical properties, and is used to calculate the pressure drops of the system. The coefficient γ depends on the pipe attributes, such as its diameter, the discharge rate and the roughness

coefficient of the material of which it is made. We used $\gamma = 0.1$ in our implementation (see Appendix B).

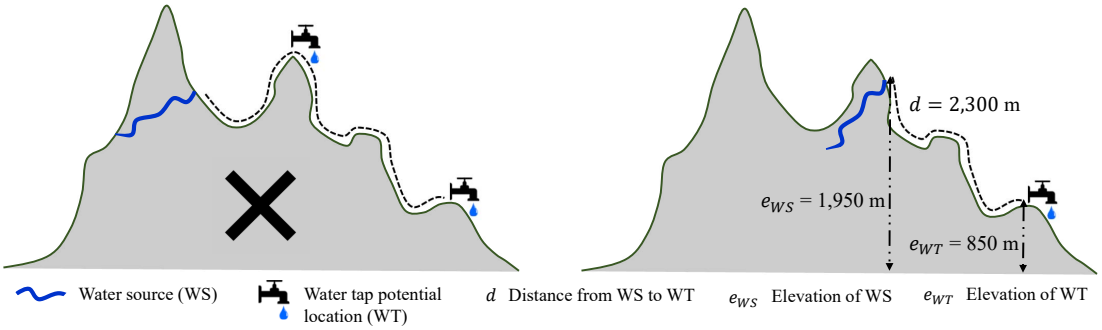

Figure 4: Infeasible and feasible connections from a WS to WTs.

Figure 4 depicts two examples of a connection from a WS to a WT. The connection to the two WTs in the example on the left does not satisfy the first technical requirement since there is a point with a higher elevation than the WS. The example on the right provides a possible connection from a WS to a WT since both requirements are respected. There is no point on the connection path with a higher elevation than the WS. In addition, considering the elevation values, the distance from WS to the potential WT location, and a value of $\gamma = 0.1$, constraint (1) is respected.

As outlined above, capacity constraints are considered. This means that the daily capacity associated to a WS must satisfy the demand from all the WTs connected to it. Since each household and each PoI is assigned to exactly one WT, each WT will have a total daily demand of water, given by the sum of the daily demands assigned to it.

The constraints imposed on the locations of WTs and the rules regarding the accessibility of households to WTs yield some simplifications. First, any potential WT location that cannot be connected to a WS due to the technical requirements cannot be used to locate a WT and is therefore eliminated. Second, every household that cannot reach any remaining WT potential location under the exceptional walking distance is eliminated. Third, a WT potential location that cannot be reached by any household is also eliminated.

3. Solution methodology

We now describe the hierarchical matheuristic we have designed for the WSSDP. Figure 5 provides an overview of the proposed solution methodology. As mentioned in Section 1.1, the

WSSDP consists of two subproblems which are solved hierarchically: the WTLAP and the WD-NDP.

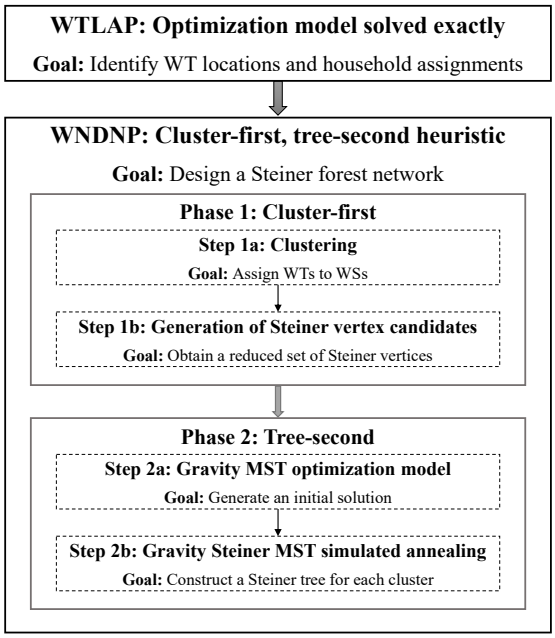

Figure 5: Structure of the matheuristic for the WSSDP.

The first subproblem consists of determining the number and the locations of WTs to install, and the allocation of the households to the WTs by assigning weights to the number of WTs and to the total user walking distances. This problem is modeled as a Fixed-Charge Facility Location Problem (Fernández and Landete, 2019) which can be solved exactly for the instance sizes considered in this paper.

The second subproblem is solved heuristically. It takes as an input the locations of WTs provided by the WTLAP solution and connects them by means of a Steiner forest subject to gravity constraints. This problem is very hard to solve, even for small-scale instances. Our computational experiments indicate that the gravity constraints are a major source of difficulty, and even relatively small instances of the GMST cannot be solved to optimality. For this reason, we cannot assess the quality of our heuristic by making comparisons with optimal solution values, as is often done.

We have developed a cluster-first, tree-second heuristic for the WDNDP. In the first phase we cluster the WTs by assigning each WT to a WS, and we identify the potential Steiner vertices

associated with each cluster. Next, in the tree phase, we design the network by heuristically computing a minimum Steiner tree for each cluster, subject to the technical requirements for a gravity-fed system. This is achieved by first optimally computing a GMST on each cluster, and then performing a local search heuristic in order to include Steiner vertices in the tree and iteratively computing an MST for each selection of Steiner vertices by means of a variation of the Prim (1957) greedy algorithm, which is easy to implement and very fast. This way of proceeding means that the local search can be performed efficiently through a single neighbourhood local search scheme consisting of the selection of Steiner vertices. There is therefore no justification to develop a sophisticated multi-operator search metaheuristic such as adaptive large neighbourhood search (Pisinger and Ropke, 2007). We have used SA which is a simple local search scheme well suited to our context. As we will show in Section 3.2.4, it uses only four parameters and is very fast.

As mentioned in Section 1.1, our methodology can easily be adapted to handle the case where only a partial network is designed. Fixing the locations of existing WTs is easily done in the WTLAP model by assigning a value of one to the corresponding decision variables. In the computation of the GMSTs, the costs of the existing pipes can be set to zero in order to promote, but not force, their inclusion in the solution.

3.1. WTLAP optimization model

The WTLAP is defined on a directed graph $G = (V, A)$, where V is the vertex set and A is the arc set. This graph is obtained by partitioning the RoI using a grid of equal-size cells and associating the centroid of each cell to a vertex of the graph. We also define the following sets, parameters, and variables:

$L \subseteq V$: set of potential WT locations;
$H \subseteq V$: set of vertices representing the cells that contain households;
$H' \subseteq L$: set of cells containing a PoI that requires a dedicated WT, $H' \cap H = \emptyset$;
$P \subset A$: set of arcs representing possible assignment of households to WTs, $(h, l) \in P$;
$P_p \subseteq P$: set of assignment arcs where arc (h, l) is within preferred distance limit;
$P_e \subseteq P$: set of assignment arcs where arc (h, l) is within exceptional distance limit;
g_h: number of households at vertex h, $h \in H$;
η: maximum number of households that can be served from the same WT;

$\widehat{d}_{hl}$: contribution that assignment (h, l) makes to the average distance. It is computed by dividing the total walking distance from h to l by the total number of households in the RoI;

α_w: weight assigned to the number of WTs installed;

α_p: weight assigned to the average walking distance in preferred cases;

α_e: weight assigned to the average walking distance in exceptional cases;

s_l: binary variable equal to one if and only if a WT is located in $l \in L$;

t_{hl}: binary variable equal to one if and only if households in $h \in H$ are assigned to the WT in $l \in L$.

The problem can then be formulated as follows:

$$(WTLAP) \quad \text{minimize } \alpha_w \sum_{l \in L} s_l + \alpha_p \sum_{(h,l) \in P_p} \widehat{d}_{hl} t_{hl} + \alpha_e \sum_{(h,l) \in P_e} \widehat{d}_{hl} t_{hl} \tag{2}$$

subject to

$$\sum_{(h,l) \in P} t_{hl} = 1 \qquad h \in H \tag{3}$$

$$\sum_{h|(h,l) \in P} g_h t_{hl} \leq \eta s_l \qquad l \in L \setminus H' \tag{4}$$

$$s_l = 1 \qquad l \in H' \tag{5}$$

$$s_l \in \{0, 1\} \qquad l \in L \setminus H' \tag{6}$$

$$t_{hl} \in \{0, 1\} \qquad (h, l) \in P. \tag{7}$$

The first term of the objective function (2) minimizes the number of installed WTs. The second and third terms represent the average walking distance associated with the WASH preferred and exceptional walking distances, respectively. Equalities (3) mean that each cell containing households is served by one WT. Constraints (4) guarantee that an opened WT in l covers at most the maximum allowed number of households assigned to that vertex, and also ensure that households cannot be assigned to a potential WT location if no WT is located in it. Equalities (5) ensure that PoIs are assigned to a dedicated WT in the same cell. In particular, constraints (5) state that a WT has to be installed at that particular vertex. Finally, constraints (6) and (7) define the domains of the variables. Note that the walking accessibility standards are implicitly satisfied since the variables t_{hl} are only defined for those pairs that fulfil the WASH standards.

3.2. WDNDP cluster-first, tree-second heuristic

The WDNDP is also defined on the directed graph $G = (V, A)$, where the following sets, parameters and variables are defined:

$L \subseteq V$: set of WTs locations defined in the (WTLAP);

$F \subseteq V$: set of WSs with associated daily capacity c_f, $f \in F$;

$L_f \subseteq L$: set of WTs assigned to WS f in the solution of the clustering process;

$R_f \subseteq V$: set of vertices representing cells that contain an access point to WS, $f \in F$;

$S \subseteq V$: set of Steiner vertices;

0: a supersource feeding all the WSs;

$O \subset A$: set of arcs (i, j) representing possible connections in the water supply network with associated distance d_{ij};

$O_{R_f L} \subset O$: set of arcs representing possible connections between WS access points of WS f, $f \in F$, and WTs;

$O_{R_f S} \subset O$: set of arcs representing possible connections between WS access points of WS f, $f \in F$, and Steiner vertices;

$O_{LS} \subset O$: set of arcs representing possible connections between WTs and Steiner vertices;

$O_{SL} \subset O$: set of arcs representing possible connections between Steiner vertices and WTs;

$O_{LL} \subset O$: set of arcs representing possible connections between WTs;

$O_{SS} \subset O$: set of arcs representing possible connections between Steiner vertices;

O_{0R_f}: set of arcs $(0, r)$ representing the connections from the supersource to an access point $r \in R_f$, $f \in F$ with associated distance $d_{0r} = 0$;

q: average daily demand for a household;

q_l: aggregation of the daily water demand of the households or PoIs assigned to WT, $l \in L \setminus H'$, i.e., $q_l = \sum_{h \in H} q g_h t_{hl}$;

q'_l: average daily demand of a PoI located in $l \in H'$;

e_i: elevation of a vertex i;

D: sum of all distances;

z_{fl}: binary variable equal to one if and only if WT $l \in L$ is assigned to WS $f \in F$;

x_{ij}: binary variable equal to one if and only if arc $(i, j) \in O_{R_f L} \cup O_{LL} \cup O_{0R_f}$ is selected;

v_{ij}: amount of water flow through arc $(i, j) \in O_{R_f L} \cup O_{LL} \cup O_{0R_f}$;

y_{rj}: binary variable equal to one if and only if the WT located at $j \in R_f \cup L_f$ receives water from a path coming from $r \in R_f$;

w_j: elevation of the source point that serves $j \in R_f \cup L_f$;

u_j: length of the path from the source point to $j \in R_f \cup L_f$.

In order to solve real-size instances of the WDNDP, we have developed a cluster-first, tree-second heuristic. In the cluster phase, an integer linear programming model is solved to group the WTs and assign them to an available WS. Then, in order to narrow down the set of Steiner vertices, three procedures are presented to determine the candidate set of Steiner vertices. Finally, in the tree phase, for each of the clusters of WTs an initial solution is obtained by solving an MST optimization model that includes the two technical requirements for a gravity-fed system (Section 2.2). Then, four variants of an SA local search heuristic, that differ from each other in the procedure developed to build the new neighbour solution, are applied to improve the initial solution. In the following subsections, we describe the proposed heuristic in detail. Section 3.2.1 presents the model implemented for grouping the WTs. Section 3.2.2 describes three procedures for identifying the Steiner vertices. Section 3.2.3 provides the mathematical programming formulation for the gravity-fed system used as an initial solution. Finally, Section 3.2.4 describes the neighbourhood search mechanism, i.e., the procedures applied to build the neighbouring solution in the SA heuristic.

3.2.1. Clustering

The assigment of WTs to WSs can be formulated and solved as the Generalized Assignment Problem (GAP). This problem computes a minimum distance assignment of WTs to capacitated WSs and can be formulated as the following integer linear program:

$$(GAP) \quad \text{Minimize} \sum_{f \in F} \sum_{l \in L} d_{fl} z_{fl} \tag{8}$$

subject to

$$\sum_{f \in F} z_{fl} = 1 \qquad l \in L \tag{9}$$

$$\sum_{l \in L} q_l z_{fl} \leq c_f \quad f \in F \tag{10}$$

$$z_{fl} \in \{0,1\} \qquad f \in F,\ l \in L. \tag{11}$$

Equalities (9) guarantee that each WT is assigned to exactly one WS. Constraints (10) limit the WTs assigned to a WS by ensuring that the daily capacity is not exceeded. Finally, the binary conditions on the decision variables are imposed through constraints (11). Note that

the variables z_{fl} are only defined for those pairs that verify the technical requirements for a gravity-fed system, meaning that WT l can be reached from WS f. With this model, we can identify $|F|$ clusters of WTs, each of which associated with one WS.

3.2.2. Generation of sets of promising Steiner vertex candidates

Referring to network trees, Steiner vertices are extra intermediate vertices that may be included in the network solution in order to connect all leaf vertices (WTs). The purpose of adding Steiner vertices is to create a shorter-length network. In our problem, all reachable cells in the RoI may become a potential Steiner vertex. We reduce the cardinality of the set S of Steiner vertices from a very large set, already discretized from a continuous space into cells and cell centroids (see Section 4.1.2), to a heuristically constructed finite set of much smaller cardinality containing promising Steiner vertex candidates. We suggest three alternative procedures to determine which cells to keep as potential Steiner vertices considering the WTs coordinates. These procedures will be compared in Section 4.3.3 and are separately applied to each of the clusters obtained during the clustering process:

1) The first procedure, called S_K, is to apply the K-means algorithm to the WT coordinates for different values of K. The K-means algorithm is a common clustering method used, for example, in data mining to partition a set of observations into K clusters in which each observation belongs to the closest midpoint. At this point, we are not interested in the partitioning result but only in the midpoints identified to become potential Steiner vertices. The steps of the algorithm are provided in Algorithm 1. The stopping criterion is set to 100 iterations, and the algorithm is iteratively applied for $K = 1, ..., |L_f|/2$, resulting in at most $[|L_f|/2(1 + |L_f|/2)]/2$ potential Steiner vertices.

Algorithm 1 Pseudocode for the K-Means algorithm.

Require: L set of data points (WTs) and number of desired partitions $|K| \geq 2$

Ensure: K set of partitions associated to a midpoint $m_1, m_2, ..., m_{|K|}$

```
iter = 0
repeat
    for all data points l ∈ L do
        Assign l to the closest m_j    1 ≤ j ≤ |K|
    end for
    Recompute the partition midpoints according to the assigned data points
    iter++
until iter = iter_max
```

2) The second procedure, called S_3, identifies one potential Steiner vertex for each subset of three WTs by finding the midpoint of the subset. The coordinates of the midpoint are computed as the mean of the WTs coordinates and replaced with its cell centroid. The number of potential Steiner vertices is then $\binom{|L_f|}{3} = |L_f|!/[3!(|L_f| - 3)!]$.

3) The third procedure, called S_F, is to use Fermat points, also called Toricelli points, as potential Steiner vertices. The Fermat point of a triangle is the point that minimizes the sum of distances between it and the three vertices of the triangle. Hence, similarly to the second procedure, for each subset of three WTs we identify the Fermat point as a potential Steiner vertex. However, if the three WTs form a triangle with an angle greater than 120°, then the Fermat point is located at one of the WTs (the obtuse-angle vertex), and therefore it does not become a potential Steiner vertex. The number of potential Steiner vertices cannot exceed $\binom{|L_f|}{3}$. The coordinates (x, y) of the Fermat point can be computed in closed form by using calculus, as proposed by Palacios-Vélez et al. (2015) (see Appendix C), and the Fermat point is replaced with its cell centroid.

3.2.3. Gravity MST optimization model

The first step in the tree phase is to solve the GMST optimization model to obtain an initial solution which will later be improved by SA. The GMST adds the technical requirements of a gravity-fed system to a classical MST model (Magnanti and Wolsey, 1995). Based on the clusters obtained during the clustering process, the gravity MST is solved separately for each WS $f \in F$ and the set L_f of its assigned WTs.

The gravity MST for a WS f can then be formulated as follows:

$$(GMST(f)) \quad \text{Minimize} \sum_{(i,j) \in O_{RR_f L} \cup O_{LL}} d_{ij} x_{ij} \tag{12}$$

subject to

$$\sum_{(i,j)\in O_{0R_f}} x_{ij} = |R_f| \tag{13}$$

$$\sum_{i|(i,j)\in O_{RL}\cup O_{LL}} x_{ij} = 1 \qquad j \in L_f \tag{14}$$

$$x_{ij} + x_{ji} \leq 1 \qquad (i,j) \in O_{LL} \tag{15}$$

$$v_{ij} \leq c_f x_{ij} \qquad (i,j) \in O_{R_f L} \cup O_{LL} \cup O_{0R_f} \tag{16}$$

$$\sum_{(0,j)\in O_{0R_f}} v_{0j} \leq c_f \tag{17}$$

$$\sum_{i|(i,j)\in O_{R_f L}\cup O_{LL}} v_{ij} - \sum_{\in O_{LL}} v_{ji} = q_j \quad j \in L_f \tag{18}$$

$$v_{0r} = \sum_{i|(r,i)} \in O_{R_f L} v_{ri} \qquad r \in R_f \tag{19}$$

$$\sum_{r\in R_f} y_{rj} = 1 \qquad \forall j \in R_f \cup L_f \tag{20}$$

$$y_{rr} = 1 \qquad \forall r \in R_f \tag{21}$$

$$w_j = \sum_{r\in R_f} y_{rj} e_r \qquad j \in R_f \cup L_f \tag{22}$$

$$u_i = 0 \qquad i \in R_f \tag{23}$$

$$u_j \geq u_i + d_{ij} - D(1 - x_{ij}) \qquad i,\ j \in R_f \cup L_f \tag{24}$$

$$w_i - e_i - 0.1u_i \geq 0 \qquad i \in L_f \tag{25}$$

$$x_{ij} \in \{0,1\} \qquad (i,j) \in O_{R_f L} \cup O_{LL} \cup O_{0R_f} \tag{26}$$

$$v_{ij} \in \mathbb{R}^+ \qquad (i,j) \in O_{R_f L} \cup O_{LL} \cup O_{0R_f} \tag{27}$$

$$y_{rj} \in \{0,1\} \qquad r \in R_f,\ j \in L \tag{28}$$

$$w_j \in \mathbb{R}^+ \qquad j \in R_f \cup L_f \tag{29}$$

$$u_j \in \mathbb{R}^+ \qquad j \in R_f \cup L_f. \tag{30}$$

The objective (12) minimizes the total length of the arcs needed to connect the WTs to the WS f. Constraints (13) and (14) limit the number of arcs to be selected. The number of arcs that form a tree is $n-1$, where n is the number of vertices. Note that the vertices in the gravity MST include the access points, the assigned WTs, and the supersource, i.e., $n = |R_f| + |L_f| + 1$. Hence, $|R_f| + |L_f|$ arcs have to be selected. Equality (13) forces the

use of all connections between the suppersource and the WS access points of WS f, while the combination of constraints (14), that guarantee an incoming arc for all WTs, forces the use of the remaining $|L_f|$ arcs. Constraints (15) ensure that at most one arc between (i, j) and (j, i) belongs to a feasible solution. Constraints (16) link the x and v variables, ensuring that there exists a flow only through the selected arcs. At the same time, for the selected arcs, constraints (16) bound the flow with the capacity of the WS f. Constraint (17) guarantees that the flow sent from WS f does not exceed its capacity. Constraints (18) and (19) are flow conservation constraints for WTs and access points, respectively, and prevent the formation of circuits. Constraints (20) mean that each vertex receives water from a unique access point. In particular, by (21) each access point receives water from itself. Constraints (22)–(25) ensure the respect of the gravity system technical requirements. Constraints (22) assign to each vertex the elevation of the access point from where it receives water. Constraints (23) and (24) assign to each vertex the distance to its access point. Note that for the access point (constraints (23)), the distance is set to zero. For WTs, constraints (24) state that if arc (i, j) is used, then the length u_j from the source to j, is the length u_i from the source to i, plus the distance d_{ij} between i and j; if arc (i, j) is not used, then the constraints are trivially satisfied. Constraints (25) impose the second technical condition (1). Finally, the domains of the decision variables are imposed through constraints (26)–(30).

3.2.4. Gravity Steiner MST simulated annealing

The second step in the tree-phase heuristic is to improve the initial solution obtained from the gravity MST model by using Steiner vertices through an SA heuristic procedure. Starting from an initial solution, at each iteration a new solution is obtained from the neighbourhood of the current solution, and the objective function value of the new solution is compared with that of the current best solution. If the new solution is better, it then becomes the current solution and the algorithm proceeds to the next iteration. If the new solution is worse, then the new solution may become the current solution with a small probability in order to avoid being trapped in a local optimum. This process is reiterated until reaching the maximum number of iterations.

We use a neighbourhood structure that consists of randomly choosing a single Steiner vertex. If the chosen Steiner vertex belongs to the current solution, we remove it from the new solution which will only consider the remaining vertices. Otherwise, we add it to the new solution. Note that the new neighbourhood solution has to be generated again to ensure its feasibility, yielding

a tree network that satisfies the gravity-fed system technical requirements. We present four procedures, which will be compared in Section 4.3.4, to rebuild the new neighbourhood solution based on Prim's algorithm for undirected MSTs (Prim, 1957). This algorithm is a greedy construction process, where from an initial vertex at each iteration the minimum cost edge that connects the vertices in the tree to those not yet in the tree is added. The MST is complete when all vertices are connected. All rebuilding procedures select the fictional supersource as an initial vertex from which all arcs to access points, $(0, r) \in O_{0R_f}$, are selected. From this point they follow different directions:

1) Procedure P1: The added arc corresponds to the shortest arc that satisfies the gravity technical requirements and ensures connectivity, i.e., it connects the vertices in the tree to those not yet in the tree.

2) Procedure P2: The added arc also satisfies the gravity technical requirements and the connectivity condition, but we distinguish whether the arc is incident or not to an access point. If so, we add the longest arc to the tree. In contrast, if the arc is not incident to an access point, we add the shortest arc.

3) Procedure P3: This procedure always adds the shortest arc, as in the Prim algorithm. Once the tree is completed, a feasibility check is performed to ensure that the gravity technical requirements are satisfied. If feasibility is not achieved, then a new iteration is performed where the inclusion of the arc that breaks feasibility is forbidden for several iterations. The iterative process ends when a feasible solution has been identified or when the maximum number of iterations has been reached.

4) Procedure P4: Apply at each iteration each of the first three procedures and select the best solution.

Algorithm 2 Pseudocode for the gravity Steiner MST simulated annealing heuristic.

Require: T_0, T_t, N_{iter}, β and $ISol$

Ensure: Network tree

```
Let T := T_0, CSol := ISol, CObj := IObj, BSol := ISol, BObj := IObj
repeat
    iter := 0
    repeat
        Generate a new neighbour solution
        if (NObj < CObj) then
            CSol := NSol, CObj := NObj
            if (NObj < BObj) then
                BSol := NSol, BObj := NObj
            end if
        else
            rand := random[0, 1], α := e^((NObj−CObj)/T
            if α > rand then
                CSol := NSol, CObj := NObj
            end if
        end if
        iter++
    until (iter ≥ N_iter)
    T := βT
until (T ≤ T_t)
```

The SA heuristic, detailed in Algorithm 2, uses four parameters: T_0, T_t, N_{iter} and β: T_0 denotes the initial temperature, T_t is the final temperature, N_{iter} represents the total number of iterations for which the neighbourhood search is applied for a given temperature, and $\beta \in [0, 1]$ is the control coefficient of the temperature reduction. At the first iteration, the current temperature T takes the value of the initial temperature T_0. The initial solution $ISol$ obtained from the gravity MST model is set as the current solution $CSol$ and the best solution $BSol$; its objective function value is set as the current objective value $CObj$ and the best objective value $BObj$. At each subsequent iteration a new neighbourhood solution $NSol$ is generated by using one of the four rebuilding procedures just described. The objective function values of the new neighbourhood solution $NObj$ and the current solution are compared. If the new objective function value is smaller, then the new solution replaces the current solution. If the objective function value of the new neighbourhood solution is also smaller than the best known objective value, then the best solution is updated with the new solution. If the value of the new objective function value is not smaller than that of the current solution, then the new

neighbourhood solution is accepted if $e^{(NObj-CObj)/T}$ exceeds a randomly generated number $rand \in [0, 1]$, and the candidate solution replaces the current solution. After N_{iter} iterations for a given temperature T, the temperature decreases according to the cooling schedule $T = \beta T$. The algorithm terminates when the current temperature T reaches the final temperature T_t.

4. Computational study

We now present the results of the computational study we have performed to analyze the proposed solution method. In Section 4.1, we describe the Nepal data set. In Section 4.2 we provide the implementation details of our approach. In Section 4.3, we present our numerical results and analyses. A discussion and the presentation of managerial insights follow in Section 4.4.

4.1. The Nepal data set

The RoI considered in this study consists of the two VDCs of Dolakha: Suspa Kshemawati and Lapilang. Dolakha (see Figure 2) was one of the districts most affected by the May 2015 earthquake. To support the rehabilitation of the community water supply systems through the design of a gravity-fed system that optimizes the access to drinking water, we gathered information on the characteristics of the desired system and on the geography of the RoI. Due to the fact that input data are obtained from several sources and in different formats, it was necessary to process and unify the available data. The Nepal dataset we used is detailed below.

4.1.1. Parameters

The information on the requirements that the water supply system has to respect was obtained from the AutRC which provided us with the identification of the available WSs and their capacity as well as the WASH accessibility and quantity standards. The assumptions made about the water supply coverage were also developed with the AutRC, based on national standards and statistics. In this context, for a basic service level, the maximum number of households that can be assigned to a WT is fixed at 21 households of five residents each. The daily amount of water required by person is 45 litres according to the Nepal WASH standards (The Government of Nepal and Republic of Finland, 2010), while for the PoIs, the daily need per person is estimated to be 15 litres, with 50 people to accommodate, translating into a daily demand of 750 litres. The AutRC aimed to build a water supply system in which people can have access to WTs in accordance to the WASH preferred horizontal and vertical walking distances between a household and a WT, fixed at 150 and 50 meters, respectively. In exceptional

cases, these standards can be widened to 250 and 80 meters. These numbers are similar to those provided in other case studies related to gravity-fed systems. For example, in their technical paper about public standpipe design for rural South Africa, Haarhoff and Rietveld (2009) state that WTs should be located at a maximum walking distance of 200 meters from a household. A more general study (Faiia, 1982) describes the basic principles of gravity flow systems and provides guidelines for design parameters which require that no more than 20% of the users walk more than 100 meters to obtain water.

A further step was to process some of these data to determine the parameter γ in (1). We set $\gamma = 0.1$ in order to be able to serve 5,000 users from a single source with a daily demand of 45 liters of water concentrated during morning hours. To this end, we considered steel pipes with an eight cm diameter (see Appendix B).

It was also necessary to identify the number of households and WSs and their coordinates (elevation and position). The elevation is crucial to ensure that the technical requirements of the gravity-fed system are satisfied. The position also plays an essential role since it provides the information needed for the distance computations. These data were extracted from satellite imagery and digital surface models by the Interfaculty Department of Geoinformatics Z-GIS of the University of Salzburg. In particular, by combining input data from satellite imagery where households and WSs were identified and with digital surface models, it is possible to obtain an attribute table of base points where the coordinates and characteristics of each element (household or WS) are recorded.

4.1.2. Grid generation

We divided the RoI using a grid of equal-size cells (Figure 6). Considering that on average the households cover an area of about 5 m $\times$ 5 m, we chose to use square cells of side 25 m. The satellite image shown in Figure 6 depicts a small area from Lapilang with the grid used to discretize the RoI. The highlighted cells are those containing at least one household or one PoI, and therefore their centroids belong to the subset $H \cup H' \subseteq V$. Analogously, the centroids of the cells crossed by the WS belong to $R_f \subseteq V$. A priori, the centroid of any cell of the grid can be a Steiner vertex or a potential WT location. We obtain graphs with 29,900 vertices and 75,200 arcs for Suspa Kshemawati and 28,500 vertices and 68,300 arcs for Lapilang.

4.1.3. Distance calculations

Note that the horizontal and vertical distances both play an important role in the graph generation. Since Nepal is a mountainous country with significant denivellations we will provide

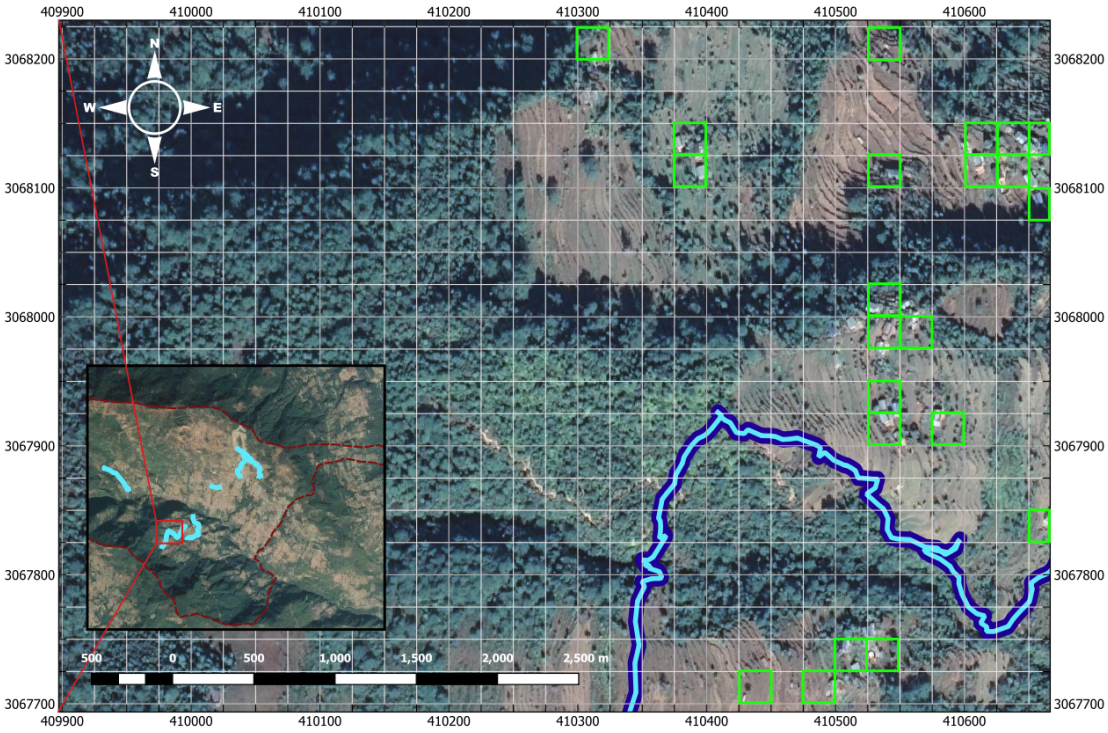

Figure 6: Satellite image partitioned into a 25 m × 25 m grid used to define the cells.

a description of how the three-dimensional distances are calculated. We have developed slightly different algorithms according to the type of distances we aim to calculate, i.e., 1) distances from WS cells, 2) distances between cells that are WT potential locations or Steiner vertices, as well as 3) distances to household cells. These algorithms are based on the classical Dijkstra's shortest path algorithm (Dijkstra, 1959).

1) Algorithm 3 summarizes the distance calculation from a WS access point $r \in R$ to $v \in L \cup S$ denoted by $d^R_{3D}(r, v)$, where R indicates that the first cell for these distances is always a WS access point. Line 10 ensures that no cell with an elevation higher than the elevation of the WS cell belongs to the path, while line 12 ensures that formula (1) is respected.

2) Algorithm 4 summarizes the calculation of the shortest distances from $v \in (L \cup S)$ to all cells $u \in (L \cup S) \setminus \{v\}$ when considering source $f \in F$, from which v and u can be reached. If a cell v can receive water from a source f but the cell u cannot, then no arc from v to u is defined since it is not possible to send water from the source f to v and then from v to u. For this reason these distances also depend on the sources; we denote by W^f the set of

Algorithm 3 3D shortest path algorithm from a source cell $r \in R$ to all cells in $L \cup S$.

1: **Input:** $G = (V, A)$, a WS access point $r \in R$, set $U_r = \emptyset$ of cells that are not reachable from r
2: **Output:** shortest distances from $r \in R$ to all cells in $L \cup S$
3: $d^R_{3D}(r, v) \leftarrow$infinity, $v \in L \cup S$
4: father[v] $\leftarrow$ undefined, $v \in L \cup S$
5: $d^R_{3D}(r, r) \leftarrow 0$
6: add r to list Q
7: **while** Q is not empty **do**
8: $v \leftarrow$ vertex in Q with the min $d^R_{3D}(r, v)$ and remove v from Q
9: **for all** u neighbour of v **do**
10: **if** $e_r \geq e_u$ **then**
11: **if** $d^R_{3D}(r, u) > d^R_{3D}(r, v) + d(v, u)$ **then**
12: **if** $e_r - e_u - 0.1(d^R_{3D}(r, v) + d(v, u)) \geq 0$ **then**
13: $d^R_{3D}(r, u) = d^R_{3D}(r, v) + d(v, u)$ and add u to Q
14: father[u] $\leftarrow v$
15: **end if**
16: **end if**
17: **end if**
18: **end for**
19: **end while**
20: **return** $d^R_{3D}(r, v)$ $r \in R,\ v \in L \cup S$

vertices unreachable from source f. Moreover, if both v and u are reachable from $f \in F$, all the cells that have to be crossed in the path from v to u must be reachable from f. Let $d^{L\cup S}_{3D}(v, u)^f$ denote the corresponding distance. Line 10 ensures that all the cells that are considered can be reached directly from the source f. If not, in line 16 these cells are added to the set T^f_v of cells that cannot be reached from v when considering source f. For all cells $u \in L \cup S \setminus \{v\}$ such that $u \notin T^f_v$, if there exists $r \in R_f$ with $f \in F$ from which it is possible to reach both v and u, and (1) is satisfied, then a new arc (v, u) belonging to the set N is defined. Note that the first condition is also satisfied since u does not belong to T_v.

3) Finally, Algorithm 5 summarizes the steps followed to calculate horizontal and vertical distances, from $h \in H$ to $v \in L$. From now on we will denote the horizontal distances by $\delta^H_{3D}(h, v)$ and the vertical distances by $\sigma^H_{3D}(h, v)$, where H indicates that the origin node is always a household cell and therefore belongs to H. Since the walking distances have both horizontal and vertical limits, in line 11 we define a weighted distance Wd that depends on both the horizontal and the vertical distance between two neighbour cells.

Algorithm 4 3D shortest path algorithm from $v \in \{L \cup S\}$ to $u \in \{L \cup S\} \setminus \{v\}$, for $f \in F$.

1: **Input:** $G = (V, A)$, $v \in \{L \cup S\}$, $f \in F$, set $T_v^f = \emptyset$ of cells that are not reachable from v, when considering source f
2: **Output:** shortest distances from $v \in \{L \cup S\}$ to all cells in $\{L \cup S\} \setminus \{v\}$ such that u can also be reached from f, set T_v^f of cells that are not reachable from v, when considering source f
3: $d_{3D}^{L\cup S}(v,u)^f \leftarrow$infinity, $u \in \{L \cup S\} \setminus \{v\}$
4: father[v] $\leftarrow$ undefined, $v \in \{L \cup S\} \setminus \{v\}$
5: $d_{3D}^{L\cup S}(v,v)^f \leftarrow 0$
6: add v to list Q
7: **while** Q is not empty **do**
8: $u \leftarrow$ vertex in Q with the min $d_{3D}^{L\cup S}(v,u)^f$ and remove u from Q
9: **for all** w neighbour of u **do**
10: **if** $u \notin W^f$ **then**
11: **if** $d_{3D}^{L\cup S}(v,w)^f > d_{3D}^{L\cup S}(v,u)^f + d(u,w)$ **then**
12: $d_{3D}^{L\cup S}(v,w)^f = d_{3D}^{L\cup S}(v,u)^f + d(u,w)$ and add w to Q
13: father[w] $\leftarrow u$
14: **end if**
15: **else**
16: add u to T_v^f
17: **end if**
18: **end for**
19: **end while**
20: **return** $d_{3D}^{L\cup S}(v,u)^f$ and T_v^f $v \in \{L \cup S\}, f \in F$

More precisely we use a parameter λ as coefficient of the vertical distance $\sigma(v,u)$ between neighbour cells v an u, which allows us to estimate how much it costs to walk one meter vertically, compared with one meter horizontally. This parameter is determined as the ratio of the preferred horizontal over vertical distances of the WASH standards. Then, $d_{3D}^H(h,v)$ represents the distance from h to v calculated using the weighted distances. Line 12 ensures that a distance is updated if the previous weighted distance from h to u was larger than the distance obtained by reaching u going through cell v. In lines 14 and 15 the horizontal and vertical distances are updated whenever the weighted distance is updated.

4.2. Model implementation

The algorithm was implemented in C++ and experiments were run on a Linux Server with two 2.5 gigahertz Intel Xeon CPU's and 32 gigabytes of memory. We used IBM CPLEX 12.7 Concert Technology with default parameters and a maximum computing time of one hour for

Algorithm 5 3D shortest path algorithm from a household cell $h \in H$ to $v \in L$.

Input: $G = (V, A)$, household cell $h \in H$
Output: horizontal and vertical distances from $h \in H$ to $v \in L$
$d^H_{3D}(h, v) \leftarrow$ infinity, $v \in L$
father$[v] \leftarrow$ undefined, $v \in L$
$d^H_{3D}(h, h) \leftarrow 0$
$\delta^H_{3D}(h, h) \leftarrow 0$, $\sigma^H_{3D}(h, h) \leftarrow 0$
add h to list Q
while Q is not empty **do**
    $v \leftarrow$ vertex in Q with the min $d^H_{3D}(h, v)$ and remove v from Q
    for all u neighbour of v **do**
        $Wd \leftarrow \delta(v, u) + \lambda\, \sigma(v, u)$
        if $d^H_{3D}(h, u) > d^H_{3D}(h, v) + Wd$ **then**
            $d^H_{3D}(h, u) = d^H_{3D}(h, v) + Wd$ and add u to Q
            $\delta^H_{3D}(h, u) = \delta^H_{3D}(h, v) + \delta(v, u)$
            $\sigma^H_{3D}(h, u) = \sigma^H_{3D}(h, v) + \sigma(v, u)$
            father$[u] \leftarrow v$
        end if
    end for
end while
return $\delta^H_{3D}(h, v)$, $\sigma^H_{3D}(h, v)$ and $d^H_{3D}(h, v)$ $h \in H,\ v \in L$

each instance to solve the models (WTLAP), (GAP) and (GMST(f)).

4.3. Results and analyses

Here we present and discuss the results of our numerical analyses under four headings: 4.3.1 Analysis of the weights given to the number of WTs and to the walking distances in the WTLAP; 4.3.2 Clustering analysis; 4.3.3 Analysis of three generation procedures for the Steiner vertex candidates; 4.3.4 Rebuilding the solution in the SA heuristic for the gravity Steiner MST.

4.3.1. Analysis of the weights given to the number of WTs and to the walking distances in the WTLAP

We first analyzed the optimal solution of the WTLAP. The values given to α_w, α_p, and α_e influence the number of WTs installed and the average distance users need to walk to fetch their drinking water. The values of these weights should reflect the priorities of the decision makers, and they should be determined after assessing their effect on several tentative solutions. Hence, to reach a well-balanced solution, it is important to evaluate the impact of the weights on the elements involved in the trade-off: the number of required WTs and the user walking

distance under the preferred and exceptional upper limits. In the following, we analyze in more detail the impacts of the weights α_w, α_p, and α_e. To this end, we solved the two instances of Suspa Kshemawati and Lapilang with 20 combinations of weight settings. Note that in some settings the exceptional distances are penalized with a large M value, the weight associated with exceptional distances is always greater than the weight associated with preferred distances.

Table 1 gives the number of WTs to open (# WT), the percentage of households covered within preferred distances (% preferred), and the average horizontal (δ) and vertical (σ) distances in meters traveled by the users for the two studied VDC. We observe that the number of WTs increases considerably when all the weight is assigned to the walking distances. For all parameter settings, the percentage of households covered within the preferred distance is always above 80%, and in most cases above 95%. However, while the number of users served within the preferred distance is more or less constant, the average distance traveled varies. As expected, the average distance traveled is lower for those settings in which more WTs are opened, and conversely the average distance traveled is higher when fewer WTs are opened.

Table 2 provides, for each VDC and distinguishing between preferred (δ_p, σ_p) and exceptional (δ_e, σ_e) cases, the maximum, minimum and average values of the average walking distance in meters computed over all settings. We recall that the preferred upper limits for the horizontal and vertical distances are $(\Delta, \Sigma) = (150, 50)$ and the exceptional upper limits for the horizontal and vertical exceptional distances are $(\bar{\Delta}, \bar{\Sigma}) = (250, 80)$. We can observe that in general, exceptional values δ_e and σ_e are more likely to be close to the upper limits.

In the remaining analysis we consider only some of the settings of Table 1, namely the four settings (α_w, α_p, α_e) = (1, 0, M), (0.5, 0.2, 0.3), (0.25, 0.75, M), and (0.25, 0.30, 0.45). Recall that the main objective of the AutRC is to minimize the number of WTs while ensuring that users are mostly covered within the preferred distance, which corresponds to the setting (1, 0, M). The minimum number of WTs is obtained for the setting (0.5, 0.2, 0.3), where exceptional distances are not penalized, and the weights associated with the number of WTs (α_w) and the walking distances ($\alpha_p + \alpha_e$) are equal. It is also interesting to analyze the setting providing the minimum average walking distance, but without exceeding 400 WTs, corresponding to (0.25, 0.75, M). Finally, we will also consider the setting (0.25, 0.30, 0.45) which gives us an intermediate result in terms of the number of WTs and of the average walking distance.

Table 1: WTLAP results for 20 weight settings.

Weight setting	Suspa Kshemawati				Lapilang			
(α_w, α_p, α_e)	# WT	% preferred	δ (m)	σ (m)	# WT	% preferred	δ (m)	σ (m)
(1.00, 0.00, M)	88	98.6	94	19	117	97.7	99	20
(0.75, 0.25, M)	94	98.6	73	14	125	97.7	67	13
(0.50, 0.50, M)	123	98.6	54	10	163	97.7	52	9
(0.25, 0.75, M)	204	98.6	32	6	253	97.7	35	6
(0.00, 1.00, M)	702	98.6	4	1	980	97.7	5	2
(0.50, 0.00, 0.50)	76	95.7	95	20	113	96.3	100	21
(0.50, 0.10, 0.40)	74	93.4	84	16	114	94.3	72	14
(0.50, 0.20, 0.30)	76	80.8	85	16	109	80.8	78	14
(0.25, 0.00, 0.75)	83	97.9	95	20	117	97.4	98	20
(0.25, 0.10, 0.65)	93	97.8	70	14	127	97.7	67	12
(0.25, 0.20, 0.55)	101	96.1	62	12	140	95.0	59	11
(0.25, 0.25, 0.50)	111	94.0	58	11	152	94.1	55	10
(0.25, 0.30, 0.45)	113	90.5	59	10	153	89.7	57	10
(0.25, 0.35, 0.40)	111	84.0	65	11	151	82.2	62	10
(0.00, 0.00, 1.00)	403	98.6	70	14	465	97.7	77	15
(0.00, 0.10, 0.90)	702	98.6	4	1	980	97.7	5	2
(0.00, 0.20, 0.80)	702	98.6	4	1	980	97.7	5	2
(0.00, 0.25, 0.75)	702	98.6	4	1	979	97.5	6	2
(0.00, 0.30, 0.70)	702	98.5	4	1	980	97.4	6	2
(0.00, 0.40, 0.60)	702	98.3	4	1	983	97.0	6	2

Table 2: Summary of the WTLAP results showing preferred and exceptional horizontal and vertical distances.

	Suspa Kshemawati				Lapilang			
Upper limit (m)	150	50	250	80	150	50	250	80
	δ_p	σ_p	δ_e	σ_e	δ_p	σ_p	δ_e	σ_e
Maximum	94	19	207	65	97	20	214	61
Minimum	2	1	185	35	2	1	191	35
Average	49	9	192	53	48	9	206	54

4.3.2. Clustering analysis

From the location of WTs previously obtained, and considering the WSs available, the WTs are clustered and assigned to a WS using the GAP model. The partition and assignment of WTs to WSs affect the length of the water distribution network. Although, the percentage of WTs assigned to each WS remains pretty much the same for the different weight settings, the weight setting has an impact on the partition. For example in Suspa Kshemawati, if we compare the partitions obtained, only the settings (0.25, 0.75, M) and (0.25, 0.3, 0.45) assign WTs to WS_0.

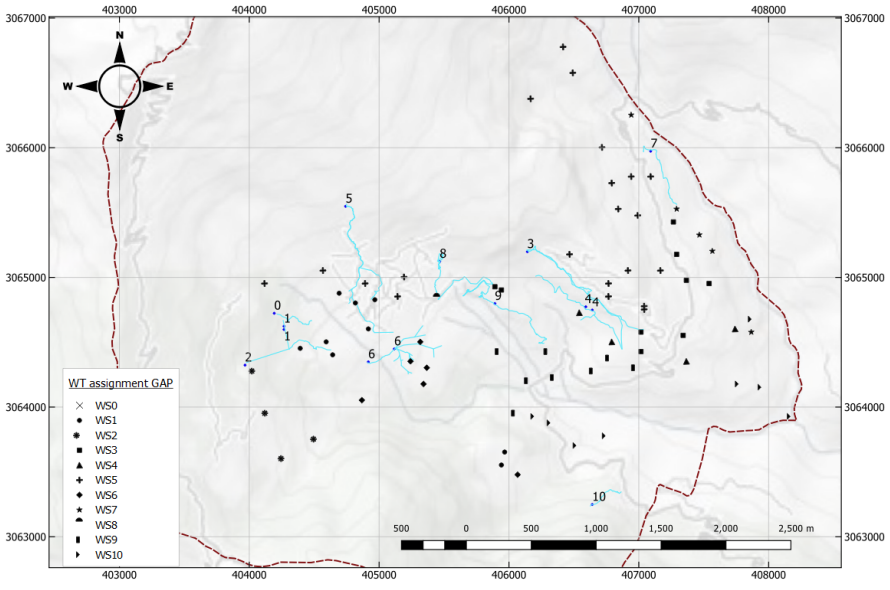

Figure 7: Cluster partitions for the Suspa Kshemawati instance (0.5, 0.2, 0.3) according to the GAP model.

Figures 7 depicts the cluster partitions obtained through the GAP model in the Suspa

Table 3: Length in meters of the water distribution network obtained by using the GAP partition.

Weight setting (α_w, α_p, α_e)	Suspa Kshemawati	Lapilang
(1, 0, M)	34298	32616
(0.5, 0.2, 0.3)	29080	30995
(0.25, 0.75, M)	41924	45987
(0.25, 0.30, 0.45)	34268	34740

Kshemawati VDC for the weight setting (0.5, 0.2, 0.3). There we see which WTs are assigned to each WS and we get an idea of the location of each of the 11 clusters. We can see how due to capacity constraints on the WSs, some WTs that are located nearby are assigned to different WSs. Consider for example the particular case of the three WTs located around the coordinates (406000, 3063500), two of them are assigned to WS_1, while the third is assigned to WS_6 and none is assigned to WS_{10} which is relatively close to these WTs.

Table 3 shows the total length of the arcs needed to connect the WTs to the WSs in the best-known solution. Note that since the GMST is solved for each WS, this table provides the sum of the tree lengths associated with each WS. It can be observed that the weight settings have the same effect on the length network in both VDC.

As mentioned before, to solve the GMST, we set a one-hour limit for each of the 72 instances (11 WSs from Suspa Kshemawati, plus seven WSs from Lapilang, multiplied by the four weight settings). We see that 87% of the instances were solved to optimality within this limit. Difficulties arose from the technical conditions of the gravity-fed system. This can be seen by analyzing the average computational time and the average gap for the nine instances not optimally solved, which are 415 seconds and 15.7%, respectively. The difficulty of our problem becomes even more apparent if we consider that the number of WTs per WS is relatively low (see Table 4). These instances would be easily solvable for the classic MST. Actually, for each of the intervals of Table 4 there are zero, one, two, three, one, and two instances not solved to optimality within the one-hour computing time limit, respectively.

4.3.3. Analysis of three generation procedures for the Steiner vertex candidates

We performed tests to analyze the three proposed procedures used to generate sets of promising Steiner vertices (Section 3.2.2). Table 5 summarizes the differences in the Steiner sets. For

Table 4: Number of instances as a function of the number of WTs per WS.

Number of WTs per WS	[0, 10]	(10, 20]	(20, 30]	(30, 40]	(40, 50]	(50, 60]
Number of instances	37	12	14	4	2	3

each VDC and each procedure, the total number of Steiner vertices found under the columns S_K, S_3 and S_F, corresponding to the K-means algorithm, every subset of three, and the Fermat point, respectively. The last three rows depicts the average, minimum, and maximum number of Steiner vertices associated with a WS. The cardinality of the Steiner sets for S_K is in the hundreds, while for S_3 and S_F it increases to thousands. To facilitate the resolution of the gravity Steiner MST, we have defined three reduced Steiner sets for S_3 and S_F by grouping them until obtaining one, three or five Steiner vertices for the WTs, denoted by $S_{3.1}$, $S_{3.3}$ and $S_{3.5}$ or $S_{F.1}$, $S_{F.3}$ and $S_{F.5}$, respectively.

Table 5: Number of Steiner vertices for three discretization methods (top) and number of WSs (bottom)

VDC	Weight setting (α_w, α_p, α_e)	S_K	S_3	S_F
Suspa Kshemawati	(1, 0, M)	143	2081	965
	(0.5, 0.2, 0.3)	103	1612	787
	(0.25, 0.75, M)	428	6289	3594
	(0.25, 0.30, 0.45)	202	3078	1780
Lapilang	(1, 0, M)	254	6012	1927
	(0.5, 0.2, 0.3)	209	4385	1601
	(0.25, 0.75, M)	716	8346	5001
	(0.25, 0.30, 0.45)	344	5898	2649
Average per WS		33	524	254
Minimum per WS		0	0	0
Maximum per WS		200	2917	2026

To assess the performance of the different Steiner sets, we compared them in terms of the length of the network, the computational time and the percentage of increase with respect to the

best average solution among the four rebuilding procedures of the gravity Steiner MST and with respect to the best solution. This analysis shows that $S_{F.1}$ provides more best solutions that any other Steiner set, four over 16. $S_{F.1}$ also requires a lower computing time, with exception of the $S_{3.1}$ which is in the same range, the average computing times of $S_{F.1}$ are are one tenth with respect the remaining sets. Furthermore, $S_{F.1}$ has the smallest increase percentage, 0.63. In contrast, S_K and $S_{3.1}$ have the highest increase, 1.43 and 1.36, respectively (see Table D1 in Appendix D).

4.3.4. Rebuilding the solution in the SA heuristic for the gravity Steiner MST

We also analyze the performance of the four proposed procedures applied to rebuild the solution during the SA heuristic in the tree-second phase (Section 3.2.4). Similarly to the previous section, the behavior of the procedures can be measured in terms of the length of the network, the computation time and the percentage of increase with respect to the average best solution or the best solution (see Table D2 in Appendix D). In terms of the length network, P4 yields the best results (eight out of 16), being the one with the smallest percentages of increase, but requires longer computing times, 15 minutes on average. In contrast, P3 provides the worst results in terms of the network length (zero out of 16), but is the fastest one, two minutes on average. P2 outperforms P1 in terms of solution quality and computing time. Finally, P4 also outperforms the other rebuilding procedures when considering only the best Steiner set $S_{F.1}$ and the reduction obtained with respect the GMST solution (see Table D3 in Appendix D). While we can see that the best solution for some instances can be obtained from different rebuilding procedures, P4 is capable of finding the best solution in seven out of the eight instances.

4.4. Discussion and managerial insights

The results of this computational study show that our proposed matheuristic based on the decomposition of the problem into two hierarchical parts is effective for two main reasons. The first reason is that it can compute good-quality solutions rather quickly, despite the very large size of the instances involved. The second reason lies in its flexibility since one can prioritize the number of WTs or the walking distance in the objective function. This is one of the advantages of solving the WT location-allocation and the water distribution network problems hierarchically. Moreover, other limiting factors can be easily added in the location of WTs or the household assignment without affecting the difficulty of the network design problem, which is already hard to solve due to the technical conditions of the gravity-fed system.

Table 6: Summary of the main decision variables.

VDC	Weight setting	WTLAP			WDNDP
	(α_w, α_p, α_e)	# WT	δ (m)	σ (m)	Network length (m)
Suspa Kshemawati	(1, 0, M)	88	94	19	33343
	(0.5, 0.2, 0.3)	76	85	16	28923
	(0.25, 0.75, M)	204	32	6	41148
	(0.25, 0.30, 0.45)	113	59	10	33930
Lapilang	(1, 0, M)	117	99	20	32503
	(0.5, 0.2, 0.3)	109	78	14	30671
	(0.25, 0.75, M)	253	35	6	45822
	(0.25, 0.30, 0.45)	153	57	10	33830

By varying the weight settings, the matheuristic yields a diversity of WTLAP solutions among which the decision maker can choose. For the Steiner set and the rebuilding procedure that performs best, Table 6 summarizes the values of the main decision variables: the number of WTs to install, the average horizontal and vertical walking distances in meters, and the length of the network in meters. We suggest implementing the solution given by the setting (0.25, 0.30, 0.45), which provides a good compromise between the network cost and the walking distance. This setting dominates the solution given by the setting (0.25, 0.75, M), and yields shorter distances than the solution given by the settings (1, 0, M) and (0.50, 0.20, 0.30) without overly increasing the length of the network. Figure 8 depicts the network obtained for Lapilang under these settings. We can observe that despite having seven potential WSs, the forest consists of 12 trees, created from different access points of the same WS. We can also see the utility of the Steiner vertices for reducing the length of the network (see the detail map at the top in Figure 8). Moreover, we can appreciate the effect that the gravity technical condition has on the topology of the network. In this sense, in the bottommost detail map we can see how the tree from the top WS access point serves some WTs that are closer to the other access point. The fact that both access points belong to the same WS indicates that these assignments are not due to the capacity constraint on the WS, but to the differences of elevations and horizontal distances.

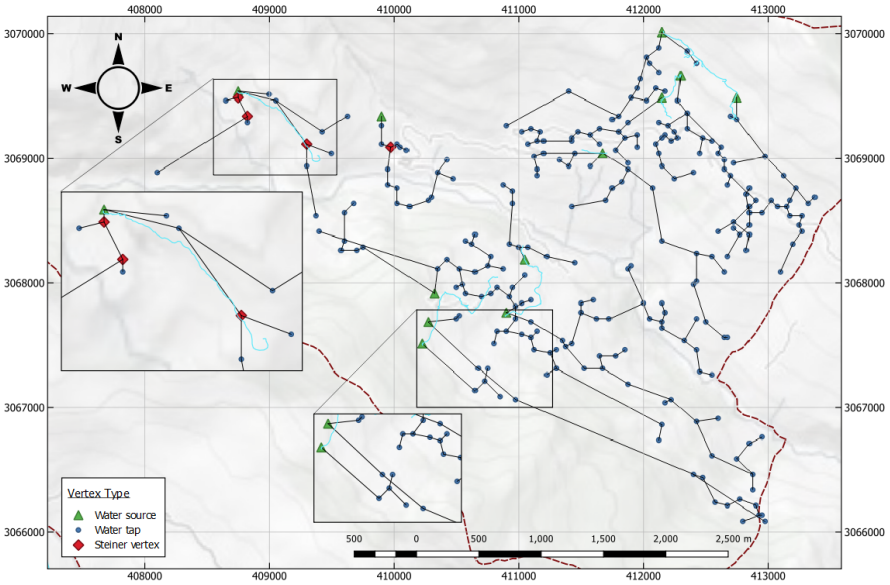

Figure 8: Network solution for instance Lapilang (0.25, 0.30, 0.45).

5. Conclusions

We have solved a network design problem motivated by the need to reconstruct the water distribution network after the 2015 earthquakes in Nepal. Gravity-fed systems with community taps are the most practical means of providing access to drinking water to populations in remote mountainous areas. It is therefore crucial to ensure access to water taps within the WASH standards, which fix the maximum horizontal and vertical walking distances to the closest water tap. We have developed a matheuristic capable of computing quickly a solution that satisfies several technical requirements. We took advantage on the hierarchical nature of the problem specified by the planners, and we solved the two subproblems sequentially. To this end, we proposed an integer linear programming model to locate the water taps and assign the households to them. We then developed a cluster-first, tree-second heuristic to design the network, with some algorithmic alternatives.

We have implemented and tested our methodology on realistic and very large-scale data sets from two Village Development Committees of Dolakha in Nepal. Despite the scarcity of data in the humanitarian sector, we succeeded in gathering data from various sources. In particular, satellite images constituted an important source of data collection. The extraction and agglomeration of all the relevant information is a difficult process that requires expertise

but leads to rich data sets. To ensure that the proposed solution adequately accounts for the mountainous character of Nepal, we have worked with distances that incorporate elevations. We have conducted extensive numerical analyses, and we have derived managerial insights that can be exploited to support implementation. Our results show the superior performance of some of the algorithmic implementation alternatives. While we have focused on a particular district of Nepal, the proposed methodology is of general applicability and can be adapted to other regions of the world.

References

Acimovic, J., Goentzel, J., 2016. Models and metrics to assess humanitarian response capacity. Journal of Operations Management 45, 11–29.

Aihara, Y., Shrestha, S., Rajbhandari, S., Bhattarai, A.P., Bista, N., Kazama, F., Shindo, J., 2018. Resilience in household water systems and quality of life after the earthquake: A mixed-methods study in urban Nepal. Water Policy 20, 1013–1026.

Ambuehl, B., Tomberge, V.M.J., Kunwar, B.M., Schertenleib, A., Marks, S.J., Inauen, J., 2021. The role of psychological ownership in safe water management: A mixed-methods study in Nepal. Water 13, 589.

Arnette, A.N., Zobel, C.W., 2019. A risk-based approach to improving disaster relief asset pre-positioning. Production and Operations Management 28, 457–478.

Bagcchi, S., 2015. Risk of infection after the Nepal earthquake. The Lancet Infectious Diseases 15, 770.

Balcik, B., Beamon, B.M., 2008. Facility location in humanitarian relief. International Journal of Logistics 11, 101–121.

Balcik, B., Bozkir, C.D.C., Kundakcioglu, O.E., 2016. A literature review on inventory management in humanitarian supply chains. Surveys in Operations Research and Management Science 21, 101–116.

Cherkesly, M., Rancourt, M.È., Smilowitz, K.R., 2019. Community healthcare network in underserved areas: Design, mathematical models, and analysis. Production and Operations Management 28, 1716–1734.

Contreras, I., Fernández, E., 2012. General network design: A unified view of combined location and network design problems. European Journal of Operational Research 219, 680–697.

D'Ambrosio, C., Lodi, A., Wiese, S., Bragalli, C., 2015. Mathematical programming techniques in water network optimization. European Journal of Operational Research 243, 774–788.

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numerische Mathematik 1, 269–271.

Dönmez, Z., Kara, B.Y., Karsu, Ö., Saldanha da Gama, F., 2021. Humanitarian facility location under uncertainty: Critical review and future prospects. Omega 102, 102393.

Doerner, K.F., Focke, A., Gutjahr, W.J., 2007. Multicriteria tour planning for mobile healthcare facilities in a developing country. European Journal of Operational Research 179, 1078–1096.

Faiia, S., 1982. Practical design notes for simple rural water systems, in: Indonesia, CARE: Attachment, p. 38.

Fernández, E., Landete, M., 2019. Fixed-charge facility location problems, in: Laporte, G., Nickel, S., Saldanha da Gama, F. (Eds.), Location Science, 2^{nd} Edition. Springer, Cham, pp. 67–98.

GC, R.K., Ranganathan, S., Hammett, A., Hall, R.P., 2021. What factors determine the technical performance of community-managed rural water systems in the middle hills of Nepal. Journal of Water, Sanitation and Hygiene for Development 11, 222–230.

Gilbert, E.N., Pollak, H.O., 1968. Steiner minimal trees. SIAM Journal on Applied Mathematics 16, 1–29.

Goldberg, M.L., 2015. Nepal earthquake facts and figures. UN Dispatch, May 2015.

Gross, J.L., Yellen, J., Zhang, P., 2013. Handbook of Graph Theory. CRC Press, Boca Raton.

Haarhoff, J., Rietveld, L., 2009. Public standpipe design and maintenance for rural South Africa. Journal of the South African Institution of Civil Engineering 51, 6–14.

Hodgson, M.J., Laporte, G., Semet, F.J., 1998. A covering tour model for planning mobile health care facilities in Suhum district, Ghana. Journal of Regional Science 38, 621–638.

Jahre, M., Kembro, J., Rezvanian, T., Ergun, Ö., Håpnes, S.J., Berling, P., 2016. Integrating supply chains for emergencies and ongoing operations in UNHCR. Journal of Operations Management 45, 57–72.

Kara, B.Y., Rancourt, M.È., 2019. Location problems in humanitarian supply chains, in: Laporte, G., Nickel, S., Saldanha da Gama, F. (Eds.), Location Science, 2^{nd} Edition. Springer, Cham, pp. 611–629.

Karsu, Ö., Kara, B.Y., Akkaya, E., Ozel, A., 2021. Clean water network design for refugee camps. Networks and Spatial Economics 21, 175–198.

Kubo, T., Yanasan, A., Herbosa, T., Buddh, N., Fernando, F., Kayano, R., 2019. Health data collection before, during and after emergencies and disasters the result of the Kobe expert meeting. International Journal of Environmental Research and Public Health 16, 893–897.

Magnanti, T.L., Wolsey, L.A., 1995. Optimal trees, in: Ball, M.O., Magnanti, T.L., Monma, C.L., Nemhauser, G.L. (Eds.), Network Models, Handbooks in Operations Research and Management Science. Elsevier, North-Holland, Amsterdam. Volume 7, pp. 503–615.

Maharjan, R., Hanaoka, S., 2017. Warehouse location determination for humanitarian relief distribution in Nepal. Transportation Research Procedia 25, 1151–1163.

Mishra, A.K., Acharya, S.R., 2018. Performance assessment of Salyankot water supply project in post-earthquake scenario of Nepal. Journal of Advanced Research in Geosciences and Remote Sensing 5, 23–40.

Naji-Azimi, Z., Renaud, J., Ruiz, A., Salari, M., 2012. A covering tour approach to the location of satellite distribution centers to supply humanitarian aid. European Journal of Operational Research 222, 596–605.

National Planning Commission, Kathmandu, Nepal 2015. Nepal earthquake 2015: Post disaster needs assessment. https://www.nepalhousingreconstruction.org.

Nepal Red Cross Society, 2017. 7th Development Plan (2016-2020). http://www.nrcs.org/sites/default/files/resources/7th%20development%20plan.pdf.

Nolz, P.C., Doerner, K.F., Gutjahr, W.J., Hartl, R.F., 2010. A bi-objective metaheuristic for disaster relief operation planning, in: Coello Coello, C.A., Dhaenens, C., Jourdan, L. (Eds.), Advances in Multi-Objective Nature Inspired Computing. Studies in Computational Intelligence. Springer, Berlin, Heidelberg, pp. 167–187.

Nomura, S., Kayano, R., Egawa, S., Harada, N., Koido, Y., 2021. Expected scopes of health emergency and disaster risk management (Health EDRM): Report on the expert workshop at the annual conference for the Japanese association for disaster medicine 2020. International Journal of Environmental Research and Public Health 18, 4447–4456.

Office for the Coordination of Humanitarian Affairs, 2015. Water, Sanitation and Hygiene (WASH): Nepal Earthquake Cluster Brief. https://reliefweb.int/sites/reliefweb.int/files/resources/20150623_wash_cluster_brief_0.pdf.

Oxfam, 2019. Water supply systems in Nepal. How to build better, more sustainable services. https://oxfamilibrary.openrepository.com/bitstream/handle/10546/620844/cs-nepal-water-supply-systems-220719.pdf/.

Palacios-Vélez, Ó.L., Pedraza-Oropeza, F.J., Escobar-Villagran, B.S., 2015. An algebraic approach to finding the Fermat-Torricelli point. International Journal of Mathematical Education in Science and Technology 46, 1252–1259.

Pisinger, D., Ropke, S., 2007. A general heuristic for vehicle routing problems. Computers & Operations Research 34, 2403–2435.

Prim, R.C., 1957. Shortest connection networks and some generalizations. The Bell System Technical Journal 36, 1389–1401.

Rancourt, M.È., Cordeau, J.F., Laporte, G., Watkins, B., 2015. Tactical network planning for food aid distribution in Kenya. Computers & Operations Research 56, 68–83.

Riedler, B., Bäuerl, M., Wendt, L., Kulessa, K., Öze, A., 2017. Remote-sensing applications to support rehabilitation of water infrastructure in Lapilang, Nepal. Journal for Geographic Information Science 1, 183–198.

Sabbaghtorkan, M., Batta, R., He, Q., 2020. Prepositioning of assets and supplies in disaster operations management: Review and research gap identification. European Journal of Operational Research 284, 1–19.

Shrestha, R., Dahal, K.R., 2020. Disaster resilient construction of water spouts in Kathmandu valley of Nepal. Journal of Civil, Construction and Environmental Engineering 5, 87–98.

Tatham, P., Pettit, S., Nolz, P.C., Doerner, K.F., Hartl, R.F., 2010. Water distribution in disaster relief. International Journal of Physical Distribution & Logistics Management 40, 693–708.

The Government of Nepal and Republic of Finland, 2010. Rural water supply and sanitation project in Western Nepal. Inception report, RWSSP-WN, 72 pages.

Trivedi, A., Singh, A., 2018. Facility location in humanitarian relief: A review. International Journal of Emergency Management 14, 213–232.

Udmale, P., Ishidaira, H., Thapa, B.R., Shakya, N.M., 2016. The status of domestic water demand: supply deficit in the Kathmandu valley, Nepal. Water 8, 196–204.

United Nations General Assembly, 2015. Sustainable development goals. `http://www.undp.org/content/undp/en/home/sustainable-development-goals.html`.

VonAchen, P., Smilowitz, K.R., Raghavan, M., Feehan, R., 2016. Optimizing community healthcare coverage in remote Liberia. Journal of Humanitarian Logistics and Supply Chain Management 6, 352–371.

Williams, G.S., Hazen, A., 1933. Hydraulic Tables: The Elements of Gagings and the Friction of Water Flowing in Pipes, Aqueducts, Sewers, etc., 3^{rd} Edition, Chapman & Hall, London.

Young, H.D., Freedman, R.A., 2020. University Physics with Modern Physics, 15^{th} Edition. Pearson Education Limited, Harrow.

Appendix A. Glossary of the abbreviations

AutRC: National Red Cross Society of Austria.

ConFLP: Connected Facility Location Problem.

GAP: Generalized Assignment Problem.

GMST: gravity minimum spanning tree.

MST: minimum spanning tree.

PoI: point of interest.

RoI: region of interest.

SA: simulated annealing.

SDG: Sustainable Development Goal.

UNICEF: United Nations Children's Emergency Fund.

VDC: Village Development Committee.

WASH: WAter, Sanitation and Hygiene.

WDNDP: Water Distribution Network Design Problem.

WHO: World Health Organization.

WS: water source.

WSSDP: Water Supply System Design Problem.

WT: water tap.

WTLAP: Water Tap Location-Allocation Problem.

Appendix B. Computation of gamma

As mentioned in Section 2.2, in order to build a gravity-fed system, some technical requirements derived from technical requirements must be satisfied. In particular, the second condition (1), based on the Hazen-Williams equation (31), relates the flow of water in a pipe with the physical properties of the pipe and the pressure loss caused by friction:

$$\gamma = 10.67 \frac{1}{\phi^{4.871}} \left(\frac{\xi}{\zeta} \right)^{1.852}, \tag{31}$$

where ξ is the pipe flow in cubic meters per second, ϕ is the pipe nominal diameter in meters, and ζ is the Hazen-Williams friction factor based on the roughness of the pipe material.

We aim to serve from a single source about 5,000 people, each with a maximum daily demand of 45 liters of water. This represents a volume of 225 m^3. We considered that the water collection is concentrated during morning time (4.5 hours). Hence, the flow computed as volume divided by time is equal to $\xi = 0.014$ m^3/s. We also considered that the diameter of the pipes is $\phi = 0.08$ m, and they are made from steel. Therefore, their friction factor is $\zeta = 130$ (Williams and Hazen, 1933), and hence $\gamma = 0.1$.

Appendix C. Computation of the Fermat point of a triangle

Consider a triangle (with no angle greater than 120°) with vertices $\mathcal{P}_a = (x_a, y_a)$, $\mathcal{P}_b = (x_b, y_b)$ and $\mathcal{P}_c = (x_c, y_c)$ (Figure C1). In our problem these three vertices correspond to WTs. Denote by a, b and c the lengths of the sides opposite $\mathcal{P}_a$, $\mathcal{P}_b$ and $\mathcal{P}_c$, respectively. Let also $\mathcal{A}$ be the triangle area. Then, as in Palacios-Vélez et al. (2015), the coordinates (x, y) of the Fermat point can be computed as

$$x = \mathcal{B}[a,b,c]x_a + \mathcal{B}[b,c,a]x_b + \mathcal{B}[c,a,b]x_c, \tag{32}$$

$$y = \mathcal{B}[a,b,c]y_a + \mathcal{B}[b,c,a]y_b + \mathcal{B}[c,a,b]y_c, \tag{33}$$

where the coefficients $\mathcal{B}[a,b,c]$, $\mathcal{B}[b,c,a]$ and $\mathcal{B}[c,a,b]$ correspond to the normalized barycentric coordinates and are computed as

$$\mathcal{B}[i,j,k] = \frac{(4\mathcal{A} + \sqrt{3}(i^2 + j^2 - k^2))(4\mathcal{A} + \sqrt{3}(i^2 - j^2 + k^2))}{8\mathcal{A}(12\mathcal{A} + \sqrt{3}(i^2 + j^2 + k^2))}. \tag{34}$$

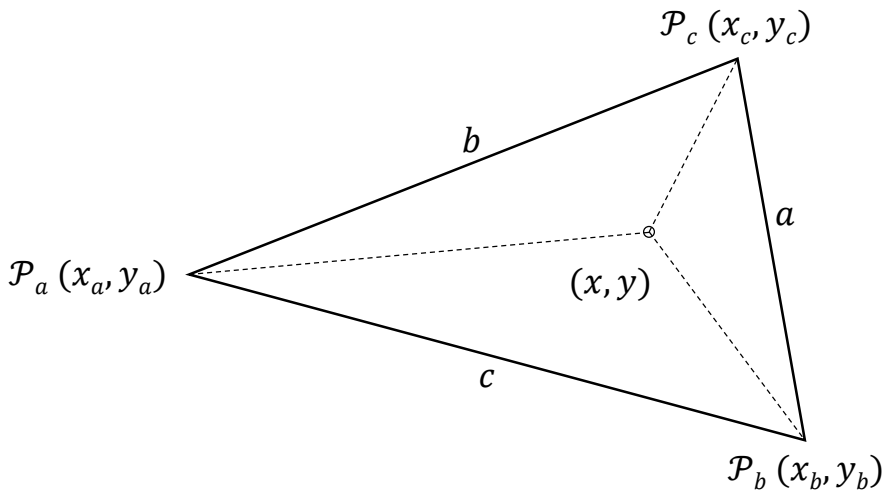

Figure C1: Fermat point for a triangle with no angle greater than 120°.

Appendix D. Steiner vertex candidates and rebuilding procedures table analysis

Tables D1 and D2 provide the length of the network, the computational time and the percentage increase with respect to the best solution. For the network length, the best solution is highlighted in bold, and for the percentage increase, we also highlight in bold the second best. Table D3 provides the network length considering only for the best Steiner set $S_{F.1}$ and the reduction obtained with respect the GMST solution.

Table D1: Comparison of Steiner sets according to the best solution.

VDC	Weight setting (α_w, α_p, α_e)	Network length (m)							Computational time (seconds)							% increase wrt best solution						
		S_K	$S_{3.1}$	$S_{F.1}$	$S_{3.3}$	$S_{F.3}$	$S_{3.5}$	$S_{F.5}$	S_K	$S_{3.1}$	$S_{F.1}$	$S_{3.3}$	$S_{F.3}$	$S_{3.5}$	$S_{F.5}$	S_K	$S_{3.1}$	$S_{F.1}$	$S_{3.3}$	$S_{F.3}$	$S_{3.5}$	$S_{F.5}$
Suspa Kshemawati	(1, 0, M)	34106	33751	33343	32972	33939	**32708**	33975	9	3	4	43	22	53	67	4.28	3.19	1.94	**0.81**	3.76	**0.00**	3.88
	(0.5, 0.2, 0.3)	28999	29050	28923	28910	28806	28980	**28772**	4	4	3	12	11	74	34	0.79	0.97	0.52	0.48	**0.12**	0.72	**0.00**
	(0.25, 0.75, M)	41612	**41092**	41146	41460	41825	41880	41594	1161	38	114	414	547	1201	3770	1.27	**0.00**	**0.14**	0.90	1.78	1.92	1.22
	(0.25, 0.30, 0.45)	34084	33919	33930	33968	**33622**	34111	33991	30	7	8	62	168	425	104	1.37	**0.88**	0.92	1.03	**0.00**	1.46	1.10
Lapilang	(1, 0, M)	**32220**	32495	32503	32254	32489	32256	32435	24	48	5	48	35	170	445	**0.00**	0.85	0.88	**0.11**	0.83	**0.11**	0.66
	(0.5, 0.2, 0.3)	30634	30933	30671	**30626**	30850	30915	30841	41	4	5	41	38	131	117	**0.02**	1.00	0.14	**0.00**	0.73	0.94	0.70
	(0.25, 0.75, M)	45953	45885	45822	45981	45624	45936	**45586**	783	83	80	930	909	2971	6699	0.81	0.66	0.52	0.87	**0.08**	0.77	**0.00**
	(0.25, 0.30, 0.45)	34580	34740	**33620**	34567	34386	34657	34388	88	12	14	149	347	417	419	2.86	3.33	**0.00**	2.82	**2.28**	3.08	2.29
Average									268	25	29	212	260	680	1457	1.43	1.36	0.63	0.88	1.20	1.13	1.23

Table D2: Comparison of the four rebuilding procedures according to the best solution.

VDC	Weight setting (α_w, α_p, α_e)	Network length (m)				Computational time (seconds)				% increase wrt best solution			
		P1	P2	P3	P4	P1	P2	P3	P4	P1	P2	P3	P4
Suspa Kshemawati	(1, 0, M)	33250	**32708**	34049	32724	53	53	3	132	1.66	**0.00**	4.10	**0.05**
	(0.5, 0.2, 0.3)	**28772**	28851	28996	28776	34	13	24	83	**0.00**	0.27	0.78	**0.01**
	(0.25, 0.75, M)	41747	**41092**	41914	41148	383	38	30	114	1.59	**0.00**	2.00	**0.14**
	(0.25, 0.30, 0.45)	33919	33891	33991	**33622**	7	71	104	168	0.88	**0.80**	1.10	**0.00**
Lapilang	(1, 0, M)	**32220**	32288	32346	32254	24	50	126	126	**0.00**	0.21	0.39	**0.11**
	(0.5, 0.2, 0.3)	**30626**	30704	30700	**30626**	41	41	4	106	**0.00**	0.25	0.24	**0.00**
	(0.25, 0.75, M)	45722	45624	45980	**45586**	2655	909	624	6699	0.30	**0.08**	0.86	**0.00**
	(0.25, 0.30, 0.45)	34388	**33620**	34467	33830	127	14	225	36	2.28	**0.00**	2.52	**0.63**
Average						415	149	143	933	0.84	0.20	1.50	0.12

Table D3: Comparison of the four rebuilding procedures for the Steiner set $S_{F.1}$.

VDC	Weight setting (α_w, α_p, α_e)	Network length (m)				% reduction wrt GMST solution			
		P1	P2	P3	P4	P1	P2	P3	P4
Suspa Kshemawati	(1, 0, M)	33990	**33343**	34049	**33343**	0.89	**2.78**	0.72	**2.78**
	(0.5, 0.2, 0.3)	**28923**	28980	29027	**28923**	**0.54**	0.34	0.18	**0.54**
	(0.25, 0.75, M)	41914	41843	41914	**41148**	0.03	0.20	0.03	**1.85**
	(0.25, 0.30, 0.45)	34553	**33930**	34141	**33930**	0.70	**0.99**	0.37	**0.99**
Lapilang	(1, 0, M)	**32503**	32537	32515	**32503**	**0.35**	0.24	0.31	**0.35**
	(0.5, 0.2, 0.3)	**30671**	30745	30700	**30671**	**1.05**	0.81	0.95	**1.05**
	(0.25, 0.75, M)	**45822**	**45822**	45987	**45822**	**0.36**	**0.36**	0.00	**0.36**
	(0.25, 0.30, 0.45)	34553	**33620**	34596	33830	0.54	**3.22**	0.42	2.62
Average						0.56	1.12	0.37	1.32

Acknowledgments

The authors are grateful to Magdalena Bauerl and Bassam Qashqo from the Austrian Red Cross, and to Lorentz Wendt from the Department of Geoinformatics Z-GIS of the University of Salzburg for several helpful discussions at the beginning of this project. Gilbert Laporte and Marie-Ève Rancourt were funded by the Canadian Natural Sciences and Engineering Research Council under grants 2015-06189 and 2014-03945. Selene Silvestri received a postdoctoral fellowship from the Institute for Data Valorisation (IVADO). Jessica Rodríguez-Pereira was funded through a postdoctoral fellowship from grants provided by IVADO and the Canada Research Chair in Distribution Management. This support is gratefully acknowledged. Thanks are also due to Joel Grau Bellet for his professional help with the geographical maps, and to the referees for their valuable comments.