Reliability problems related to path-shaped facility location on trees

Justo Puerto[1], Federica Ricca[2], Andrea Scozzari[3]
IMUS, Instituto de Matemáticas de la Universidad de Sevilla
Calle Tarfia sn. 41012, Sevilla (Spain)
puerto@us.es
[2]Sapienza Universitá di Roma (Italy)
federica.ricca@uniroma1.it
Università degli Studi "Niccolò Cusano", Roma, (Italy)
andrea.scozzari@unisucano.it

Abstract

In this paper we study a location problem on networks that combines three important issues: 1) it considers that facilities are extensive, 2) it handles simultaneously the location of more than one facility, and 3) it incorporates reliability aspects assuming that facilities may fail. The problem consists of locating two path-shaped facilities minimizing the expected service cost in the long run, assuming that paths may become unavailable and their failure probabilities are known in advance. We show that the problem in general networks is NP-hard. Then, we provide a recursive algorithm that solves the problem on tree networks in $O(n^2)$ time, where n is the number of vertices.

Keywords: Reliable location, path-shaped facilities, network location.

1 Introduction

One of the most important strategic decisions in the design of infrastructures is the location of facilities. This has motivated a lot of research on different facility location problems during many years (see, e.g., [6]) and, in particular, on several extensive path-or tree-shaped facility location problems (see, e.g., [15, 16] and the references therein).

There are several sources of uncertainty that must be considered when facing a location problem. Costs, customer demands or production capacities may be unknown at the moment of making a decision, but also unexpected events may disrupt the operation of the facilities themselves. This gives rise to situations where some facilities become temporary unavailable to provide service to customers due to system failures, natural disasters, terrorist attacks, labor strikes, etc.. Therefore, modeling issues should handle as best as they can these unknown or unpredictable situations whenever they occur.

Realistically, no decision-maker would accept a solution with very high operating costs just to hedge against very rare facility disruptions unless high penalties must be paid to costumers in case of uncovered service. Failures typically results in extra transportation costs as customers originally served by closest facilities must be redirected to more distant ones (see, e.g., [7]). In order to balance the normal and failure operation costs, the location of the facilities should depend on how likely the facilities may get disrupted, as well as on their closeness to the potential customers. This has motivated an alternative approach to the "customer-to-closest facility" cost that consists of locating facilities that minimize the total expected service cost in the long run, assuming that failures are accidental, and their probabilities can be estimated in advance [2]. Needless to say, this approach is not unique and another way to model disruption is to consider it motivated by intentional attacks [3, 12, 20]. Nevertheless, although these models are very interesting they are beyond the scope of this paper.

The literature in the area of reliable location models can be traced back, at least, to the paper [5] where both a median and a center objective functions are considered, and the situation where a fixed number of facilities might fail is tackled. However, one can observe that the current interest in reliability issues in location has been recently restarted with [18] and other works by the same authors. In addition, the following papers [2], [21], [4] and [17] have also contributed decisively to this increasing interest. On the other hand, robustness analysis of transportation networks has been widely analyzed in the literature from alternative points of view. For instance, [10] considers that a railway network is robust when passengers have several options to reach their destination. In [1] and [19] robustness of a transportation network is defined with respect to uncertainty in the origin-destination matrix. Yet another line of research on handling robustness aspects of transportation networks relates to Game Theory. Here the reader is referred to [11] for up to date references and to [8] for a review of applications.

As can be seen from the literature review there are models that consider reliability aspects of point location and some other models that apply integer programming tools to design routes but one does not find reliability models for the simultaneous location of extensive facilities. The goal of this paper is to combine three crucial aspects of location models: the existence of more than one service facility, the assumption of the extensive nature of service facilities (frequently more realistic than the assumption of locating points), and the minimization of the total expected service cost in the long run, assuming that failures are accidental and probability of failure are known. Thus, we solve the problem of finding simultaneously in a tree network two path-shaped facilities that may fail with given probabilities in order to minimize the expected total service cost. Here we assume that each customer is first assigned to its closest facility, then, if this fails, to the second closest, and if both facilities fail, he/she is assigned to a backup facility modeled by a penalty cost. We show that this problem is NP-hard on general cyclic networks. The case is different for tree networks. Assuming that the tree has n vertices, the problem can be solved by brute force by evaluating the objective function on each of the $O(n^4)$ different pairs of paths of the tree. Since each evaluation can be done in linear time, this approach would lead us to an overall complexity of $O(n^5)$ time. In spite of that, we present in this paper an $O(n^2)$ time complexity algorithm for solving this problem. It is also important to remark that our complexity result for locating two paths equals the one obtained by [2] for the corresponding two points location problem.

The paper is organized as follows. In Section 2 we provide some notation and definitions, while basic properties are introduced in Section 3. Section 4 illustrates the algorithm for solving the two path-shaped facilities location problem with probabilities of failure. In Section 5 we discuss further extensions and draw some conclusions.

2 Notation and Definitions

Let $T = (V, E)$ be a tree with $|V| = n$. Suppose that a positive real length $\ell(e) = \ell(u, v)$ is assigned to each edge $e = (u, v) \in E$. Let $A(T)$ denote the continuum set of the points in the edges of T. Each subgraph of T is also viewed as a subset of $A(T)$. The edge lengths induce a distance function on $A(T)$ so that for any pair of points x and y in $A(T)$ (i.e., vertices or points in the interior of an edge) $d(x, y)$ is the length of the (unique) path $P(x, y)$ from x to y. Then, $A(T)$ is a metric space with respect to such distance function [9]. In the following, we will avoid to specify one or both the endpoints of a path when it is not necessary. When T is rooted at a vertex r it is denoted by T_r. We denote by $V(T_r)$ the set of vertices of T_r. For any vertex v, let T_v be the subtree of T_r rooted at vertex v, $S(v)$ the set of children of v in T_r, and $p(v)$ the parent of v in T_r. Clearly, a vertex v is a leaf if and only if $|S(v)| = 0$. A path P is *discrete* if both its endpoints are vertices of T, otherwise it is *continuous*. We denote by $V(P)$ the set of vertices belonging to P. Let $d(u, P)$ be the distance from a vertex u to a path P, that is, the length of the shortest path from u to a vertex or an endpoint of P.

Given a weighted tree $T = (V, E)$, the *2 Unreliable Median Paths* (2UMP) problem consists of locating two path-shaped facilities in T, $\mathcal{P}^1$ and $\mathcal{P}^2$, each characterized by a given probability of disruption, say, p_1 and p_2, respectively. For a given location of the two facilities, that we denote by $L = \{L(\mathcal{P}^1), L(\mathcal{P}^2)\}$, a client v is first associated to its closest facility, but, in case of disruption, he/she is re-directed to the other one, and, if this also fails (meaning that no operating facilities are available to serve client v), a fixed positive penalty is applied in order to take into account the cost for the dissatisfaction of client v. For each vertex $v \in V$, let $0 \leq h_v \leq 1$ be a weight representing the fraction of the total population in v, and β_v be the fixed non negative penalty to be paid when v is not served. Since β_v is considered as the cost to serve a client in v from a facility outside the network when all the located facilities fail, we impose $\beta_v \geq \max\limits_{u \in V} d(u, v)$. This is a natural extension of the problem studied in [2] to the case of locating two paths and assigning to each client both a first and a second facility.

We denote by L^v_k, with $k = 1, 2$ the k-th closest facility to v, so that, if, for example, for a vertex v, $\mathcal{P}^2$ is the closest facility and $\mathcal{P}^1$ the second closest, one has $L^v_1 = \mathcal{P}^2$ and $L^v_2 = \mathcal{P}^1$.
To evaluate the general objective function for a given path location L, for each vertex v in V, we compute the total expected weighted cost to serve v with L:

$$Z_v[L] = h_v \left[d(v, L^v_1)\ (1 - p_{L^v_1}) + d(v, L^v_2)\ p_{L^v_1}(1 - p_{L^v_2})\right] + h_v p_{L^v_1} p_{L^v_2}\ \beta_v \tag{1}$$

Then, the objective function of 2UMP is

$$Z[L] = \sum_{v \in V} h_v \left[d(v, L^v_1)\ (1 - p_{L^v_1}) + d(v, L^v_2)\ p_{L^v_1}(1 - p_{L^v_2})\right] + \sum_{v \in V} h_v p_{L^v_1} p_{L^v_2}\ \beta_v \tag{2}$$

and the problem can be stated as follows: find in T a location L of two facilities $\mathcal{P}^1$ and $\mathcal{P}^2$ such that (2) is minimized.

We observe that on general networks 2UMP is NP-Hard since it contains the median path location problem. In fact, this is a special case of 2UMP when $p_1 = 0$ and $p_2 = 1$ or viceversa.

The key-aspect in the above 2UMP problem is concerned with the relation between the distance function and the location L of the two facilities with different probabilities of disruption. Actually, for each client v the distance function induces a complete order $\prec_v$ on the set L stating which facility is assigned to v first, that, for the two facility case, simply means that $\mathcal{P}^1 \prec \mathcal{P}^2$ if and only if $d(v, L(\mathcal{P}^1)) < d(v, L(\mathcal{P}^2))$. When the two facilities are equidistant from v the tie can be broken arbitrarily, but, since for the computation of the objective function we need to distinguish which facility is assigned to v as the closest one, in the rest of the paper we will assume that when $d(v, L(\mathcal{P}^1)) = d(v, L(\mathcal{P}^2))$, the first facility for v is always the one with lower probability of disruption, and, if this still produces a tie, we establish that the first facility for v is $\mathcal{P}^1$.

3 Basic properties and problem structure

As widely discussed in [2], in location of unreliable facilities a main issue is *co-location* which, for the point location problem corresponds to the possibility of locating different facilities at the same point of the network. Extending this notion to path-shaped facilities means that the two optimal (different) facilities may correspond to the *same* path. However, in path location an additional aspect must be discussed related to the possibility that the two optimal paths may partially intersect. Actually, the intersection between two paths leads to some important issues characterizing the structure of the problem. We provide a first remark on the existence of solutions for the problem.

Remark 1 *When intersection between paths is not allowed the optimal solution of 2UMP may not exist.*

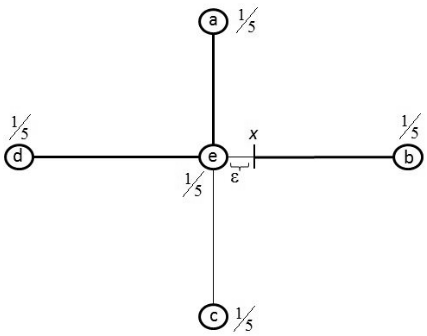

Figure 1: An example when an optimal solution of 2UMP does not exist if intersection between paths is not allowed.

Consider the tree T shown in Figure 1 with five vertices, a, b, c, d and e. Suppose that all edge lengths are equal to 1, and $h_a = h_b = h_c = h_d = h_e = \frac{1}{5}$. Consider the two disjoint paths P_{ad} and P^{ε}_{xb} where x is a point along edge (e, b) located at distance $\varepsilon > 0$ from vertex e. Consider the location of facility $\mathcal{P}^1$ in P_{ad} and facility $\mathcal{P}^2$ in P^{ε}_{xb}, i.e., $P_{ad} = L(\mathcal{P}^1)$ and $P^{\varepsilon}_{xb} = L(\mathcal{P}^2)$, with the corresponding objective function value given by:

$$\begin{aligned} Z^{\varepsilon}[L] &= \tfrac{1}{5}\, p_1(1-p_2)\,[3+4\varepsilon] \\ &+ \tfrac{1}{5}\,(1-p_1)(1+p_2) + \tfrac{1}{5}\, p_1p_2[\beta_a+\beta_b+\beta_c+\beta_d+\beta_e]. \end{aligned} \tag{3}$$

When $\varepsilon \to 0$, $Z^{\varepsilon}[L]$ decreases but it never reaches its infimum if the intersection between P_{ad} and P^{ε}_{xb} is forbidden. On the other hand, if intersection is allowed, the optimal solution of the above problem is given by $L^*(\mathcal{P}^1) = P_{ad}$ and $L^*(\mathcal{P}^2) = P_{cb}$ with objective function value

$$Z[L^*] = \frac{2}{5}\, p_1(1-p_2) + \frac{2}{5}\, p_2(1-p_1) + \frac{1}{5} p_1p_2[\beta_a+\beta_b+\beta_c+\beta_d+\beta_e]. \tag{4}$$

This example shows that there are some instances of 2UMP that, when intersection is not allowed, are not well defined, implying that the optimal solution of 2UMP might not exist.

It also shows that the intersection between the two optimal paths does not necessarily imply that they are also co-located. Actually, in the previous example the best co-located pair of paths is given, for example, by $L(\mathcal{P}^1) = L(\mathcal{P}^2) = P_{ac}$ with

$$Z[\bar{L}] = \frac{2}{5}\, p_1(1-p_2) + \frac{2}{5}\, (1-p_1) + \frac{1}{5} p_1 p_2 [\beta_a + \beta_b + \beta_c + \beta_d + \beta_e]. \tag{5}$$

Proposition 1 *In an optimal solution of 2UMP the facilities always correspond to discrete paths.*

Proof:
Let $L(\mathcal{P}^1) = P(u,v)$ and $L(\mathcal{P}^2) = P(x,d)$ be two facilities such that $\mathcal{P}^1$ is a discrete path, while $\mathcal{P}^2$ is a continuous path, d being a vertex of T, and the other endpoint x belonging to the interior of an edge (a,b). For $\mathcal{P}^2$, we evaluate the possibility of replacing $P(x,d)$ with the discrete path $P(a,d)$. For a vertex z in T, two cases may arise: i) the order induced by the distance function on the set of the two facilities for client z does not change when $P(a,d)$ replaces $P(x,d)$; ii) $P(x,d)$ was the second closest facility for z, but, after the replacement, $P(a,d)$ becomes the first one. The only relevant case is ii), since in case i) for a client z either no distance changes, or the cost to serve z strictly decreases (see, formula (1)), and thus $P(a,d)$ provides a better solution than $P(x,d)$.
In case ii), assume w.l.o.g that vertex $b \in V(P(x,d))$. One has $d(z,a) < \min_{w \in V(P(x,d))} d(z,w)$ and $d(z,P(x,d)) = d(z,a) + d(a,x)$. For client z the following holds:

$$d(z,P(u,v)) \leq d(z,P(x,d)) \quad \text{and} \quad d(z,P(u,v)) > d(z,a) = d(z,P(a,d)).$$

The cost to serve z when $L_1^z = P(u,v)$ and $L_2^z = P(x,d)$ is:

$$h_z d(z,P(u,v))(1-p_1) + h_z d(z,P(x,d)) p_1 (1-p_2) + h_z p_1 p_2 \beta_z \tag{6}$$

while, when $L_1^z = P(a,d)$ and $L_2^z = P(u,v)$, it is:

$$h_z d(z,P(a,d))(1-p_2) + h_z d(z,P(u,v)) p_2 (1-p_1) + h_z p_1 p_2 \beta_z \tag{7}$$

The difference between (6) and (7) is

$$h_z[d(z,P(u,v)) - d(z,a)](1-p_1)(1-p_2) + h_z d(a,x) p_1 (1-p_2) > 0,$$

showing that the objective function (2) strictly decreases when facility $\mathcal{P}^2$ is located in the discrete path $P(a,d)$ instead of in $P(x,d)$.

Proposition 2 *In an optimal solution of 2UMP the two optimal paths always connect two leaves of T.*

The proof follows from Prposition 1.

4 Solution algorithm for the discrete 2UMP

In [2] the problem of locating two facilities in a tree under reliability issues is already addressed, but in that case the facilities correspond to points. The authors show that a "nodal optimality" property holds if co-location is allowed, that, in their case, corresponds to locating the two facilities at the same point. The solution algorithm proposed is then based on searching separately for the best co-located solution and the best disjoint one. Here we follow a similar approach for 2UMP, taking into account that in our case, the optimal pair of paths may be either (partially) intersecting or vertex-disjoint. We describe an *ad hoc* procedure for finding the best pair of intersecting paths in Section 4.1, while, in Section 4.2 we show that an approach similar to the one in [2] can be adapted to deal with the location of two vertex-disjoint paths.

4.1 Search for intersecting paths for 2UMP

Under the assumption that the two optimal paths intersect, they share a common section, i.e., they intersect in at least one vertex r. Relying on this simple observation, the general strategy of the procedure to obtain pairs of intersecting paths is to start from one common vertex, trace out the common part first, and then generate the rest of the two paths independently. The necessary computations are facilitated if the common vertex r can be assumed to be the root of a binary tree having two children, r_1 and r_2. Actually, in this case the two paths must share the whole subpath $P(r_1, r_2)$ and, therefore, they can be built by visiting top-down T_{r_1} and T_{r_2} independently. Notice that, going down in T_{r_1} and in T_{r_2}, the two paths will coincide from r up to some vertex u which may be even a leaf. Figure 2, a) and b), shows two possible configurations of intersecting paths on a binary rooted tree sharing the subpaths $P(r_1, u)$ and $P(r_1, r_2)$, respectively.

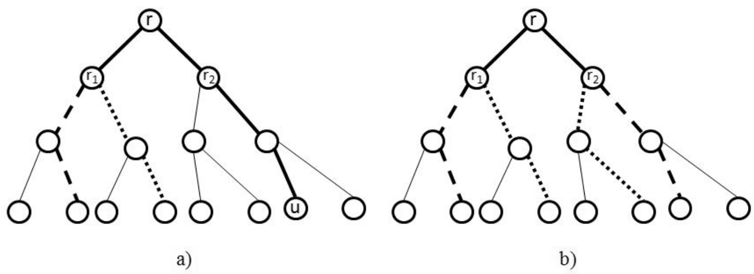

Figure 2: a) Two intersecting paths sharing the subpath $P(r_1, u)$. They separate only in vertex r_1 (see, the dashed and dotted lines), while they intersect in the subtree T_{r_2} from r_2 up to the leaf u. b) Two intersecting paths sharing the subpath $P(r_1, r_2)$. They separate both in vertex r_1 and r_2.

In order to evaluate all pairs of intersecting paths in T we consider one vertex r of T at a time, root T at r and transform T_r into a binary rooted tree by applying the procedure provided in [9]. For a given node $v \in T_r$ we denote by v_1 and v_2 its left and right child, respectively. W.l.o.g., when a node v has only one child, we always consider it as the left child of v.

For a given root r we define a pair of *best paths* P_r^1 and P_r^2 as those that, among all pairs of paths intersecting in $P(r_1, r_2)$, provide the minimum objective function value. In order to find P_r^1 and P_r^2 along with the corresponding objective function value $Z^r(L)$, with $L = \{L(P_r^1), L(P_r^2)\}$, we visit T_r top-down. The procedure relies on the computation of *saving functions* which at each stage of the visit provide the gain in the objective function (2) that can be obtained by extending the paths in a subtree.

For the sake of simplicity, we assume w.l.o.g. that $p_1 < p_2$, so that the two facilities $\mathcal{P}^1$ and $\mathcal{P}^2$ are univocally identified by their probabilities of disruption p_1 and p_2, respectively, and, as stated before, if a tie occurs for a client v, he/she is assigned to $\mathcal{P}^1$ first.

Consider the binary rooted tree T_r. In the top-down visit of T_r three different situations may arise at a given vertex v:

1. The two paths followed the same track up to vertex v, but they separate after v. Two cases are possible depending on which path passes through the left and the right child of v, v_1 and v_2, respectively.

2. The two paths followed the same track up to vertex v and, after v, they proceed together towards either v_1, or v_2. Also here we have two cases, depending on towards which son of v the two paths proceed.

3. The two paths followed the same track up to some ancestor of v so that, just one of them passes through v and proceeds into T_v.

To cope with the three above cases, we associate to each vertex v of T_r six quantities labeled as follows:

1. $S^{p_1p_2}_{\curlywedge}(v)$ is the maximum saving in the objective function when the path with probability of disruption p_1 passes through the left child of v, i.e., vertex v_1, ending in a leaf of T_{v_1}, while the path with probability of disruption p_2 passes through the right child of v, i.e., vertex v_2, and ends in a leaf of T_{v_2}. A similar quantity $S^{p_2p_1}_{\curlywedge}(v)$ is defined for the opposite case.

2. $S^{v_1}_{\parallel}(v)$ is the maximum saving in the objective function when the two paths proceed together towards v_1, ending in some leaves of T_{v_1}. A similar quantity $S^{v_2}_{\curlywedge}(v)$ is defined when the two paths proceed together towards v_2 ending in some leaves of T_{v_2}.

3. $BS^{p_1}(v)$ is the maximum saving in the objective function when only path $\mathcal{P}^1$ extends from v up to a leaf of T_v. A similar quantity $BS^{p_2}(v)$ is defined for $\mathcal{P}^2$ w.r.t. vertex v.

The above quantities can be computed recursively during a bottom-up visit of T_r.

Let $H_v = \sum\limits_{u \in V(T_v)} h_u$ be the sum of the weights of vertices in T_v for which the bottom-up computation is well-known and straightforward [13, 14].

The quantities $BS^{p_i}(v)$, $i = 1, 2$ can be computed applying the following recursive formulas:

$$BS^{p_i}(v) = \begin{cases} 0 & \text{if } v \text{ is a leaf} \\ \\ \max\{BS^{p_i}(v_1) + H_{v_1} d(v_1, v)(1 - p_i); & \\ \qquad BS^{p_i}(v_2) + H_{v_2} d(v_2, v)(1 - p_i)\} & \text{otherwise.} \end{cases} \tag{8}$$

The quantities $S^{p_ip_j}_{\curlywedge}(v)$, $i, j = 1, 2$, $i \neq j$ are computed as follows:

$$S_{\curlywedge}^{p_i p_j}(v) = \begin{cases} 0 & \text{if } v \text{ is a leaf} \\ \\ H_{v_1} d(v_1, v)(1 - p_i) + BS^{p_i}(v_1) & \\ +H_{v_2} d(v_2, v)(1 - p_j) + BS^{p_j}(v_2) & \text{otherwise.} \end{cases} \tag{9}$$

The quantities $S_{\parallel}^{v_i}(v)$, $i = 1, 2$, are computed as follows:

$$S_{\parallel}^{v_i}(v) = \begin{cases} 0 & \text{if } v \text{ is a leaf} \\ \\ H_{v_i} d(v_i, v)[(1 - p_1) & \\ +p_1(1 - p_2)] + MS(v_i) & \text{otherwise.} \end{cases} \tag{10}$$

where $MS(v_i) = \max\{S_{\curlywedge}^{p_1 p_2}(v_i),\ S_{\curlywedge}^{p_2 p_1}(v_i),\ S_{\parallel}^{v_1^i}(v_i),\ S_{\parallel}^{v_2^i}(v_i)\}$, v_1^i and v_2^i being the two children of v_i.

The objective function value associated to the pair of best paths P_r^1 and P_r^2 that intersect at least in the subpath $P(r_1, r_2)$ is then given by:

$$\begin{aligned} Z^r(L) = & \sum_{u \in V(T(r_1))} h_u d(u, r_1)[(1 - p_1) + p_1(1 - p_2)] \\ & - \max\{S_{\curlywedge}^{p_1 p_2}(r_1), S_{\curlywedge}^{p_2 p_1}(r_1), S_{\parallel}^{v_1^{r_1}}(r_1), S_{\parallel}^{v_2^{r_1}}(r_1)\} \\ & + \sum_{u \in V(T(r_2))} h_u d(u, r_2)[(1 - p_1) + p_1(1 - p_2)] \\ & - \max\{S_{\curlywedge}^{p_1 p_2}(r_2), S_{\curlywedge}^{p_2 p_1}(r_2), S_{\parallel}^{v_1^{r_2}}(r_2), S_{\parallel}^{v_2^{r_2}}(r_2)\}. \end{aligned} \tag{11}$$

Among all possible pairs of best paths P_r^1 and P_r^2 that can be obtained by rooting the tree at the different vertices $r \in V$, the best location L of two intersecting paths $\mathcal{P}^1$ and $\mathcal{P}^2$ is given by the pair with the minimum service cost (2).

If one wants to provide the structure of the two paths, it suffices to trace back the sequence of choices made in the bottom-up computation of $S_{\parallel}^{r_1}(r)$ and $S_{\parallel}^{r_2}(r)$.

4.2 Search of disjoint paths for 2UMP

In this section we discuss the case of a solution given by two paths with vertex-disjoint sets. As in [2] the approach developed in this section relies on the following basic observation. Suppose that the optimal solution of 2UMP on T corresponds to two vertex-disjoint paths $\bar{L} = \{\bar{L}(\mathcal{P}^1), \bar{L}(\mathcal{P}^2)\}$. Then it is always possible to find in T an edge (i, j) such that when it is removed from T the following holds for the two generated subtrees T_{ij} and T_{ji} containing vertex i and vertex j, respectively:

- $\bar{L}(\mathcal{P}^1)$ is in T_{ij} and $\bar{L}(\mathcal{P}^2)$ is in T_{ji} (or viceversa);
- all the vertices in T_{ij} are closer to $\bar{L}(\mathcal{P}^1)$ than to $\bar{L}(\mathcal{P}^2)$ and all the vertices in T_{ji} are closer to $\bar{L}(\mathcal{P}^2)$ than to $\bar{L}(\mathcal{P}^1)$ (or viceversa).

The idea of the procedure is to find the optimal pair $\bar{L} = \{\bar{L}(\mathcal{P}^1), \bar{L}(\mathcal{P}^2)\}$ by searching for the two paths separately in the two subtrees T_{ij} and T_{ji}. In order to do this, we follow here a similar approach to the one presented in [2] for the point location problem and extend the basic results to our 2UMP.

Discarding the constraint that in 2UMP a client must be served first by its closest facility and secondly by the other, one can state a variant of the 2UMP consisting of searching for an edge (i, j) in T and two vertex-disjoint paths, one located in T_{ij} and the other in T_{ji}, such that the following objective function is minimized:

$$\begin{aligned} & F[\{L(\mathcal{P}^1), L(\mathcal{P}^2)\}|(i,j)] = \\ & = \sum_{v \in V(T_{ij})} h_v \; d(v, L(\mathcal{P}^1)) \; (1 - p_{\mathcal{P}^1}) + \\ & \sum_{v \in V(T_{ij})} h_v \; d(v, L(\mathcal{P}^2)) \; (1 - p_{\mathcal{P}^2}) p_{\mathcal{P}^1} + \\ & + \sum_{v \in V(T_{ji})} h_v \; d(v, L(\mathcal{P}^2)) \; (1 - p_{\mathcal{P}^2}) \\ & + \sum_{v \in V(T_{ji})} h_v \; d(v, L(\mathcal{P}^1)) \; (1 - p_{\mathcal{P}^1}) p_{\mathcal{P}^2} + \\ & + \sum_{v \in V} h_v p_{\mathcal{P}^1} p_{\mathcal{P}^2} \; \beta_v \end{aligned} \tag{12}$$

The above expression corresponds to force a client in vertex v to be served first by the facility located in the same subtree where v lies. Thus, the optimization problem is:

$$\min_{(i,j) \in E} F[(i,j)] := \min_{\substack{L(\mathcal{P}^1) \in T_{ij} \\ L(\mathcal{P}^2) \in T_{ji}}} F[\{L(\mathcal{P}^1), L(\mathcal{P}^2)\}|(i,j)]. \tag{13}$$

Problem (13) is different from 2UMP because here the customer's first facility is not necessarily the closest to him/her. However, one can show that in the optimal solution of (13) the located paths coincide with the optimal paths for 2UMP, provided that the optimal solution of 2UMP is given by two vertex-disjoint paths. This result follows by a direct generalization of Theorem 5 in [2] presented below (we do not report the proof since it is basically the same as the one of Theorem 5 in [2]).

Theorem 1 *Let $\bar{L} = \{\bar{L}(\mathcal{P}^1), \bar{L}(\mathcal{P}^2)\}$ be a pair of vertex-disjoint paths corresponding to an optimal solution to 2UMP. Let (i^*, j^*) be the edge that minimizes $F[(i,j)]$ and $L^* = \{L^*(\mathcal{P}^1), L^*(\mathcal{P}^2)\}$ be the two paths minimizing $F[\{L(\mathcal{P}^1), L(\mathcal{P}^2)\}|(i^*, j^*)]$.*
Then one has $Z[\bar{L}] = F[\{L^*(\mathcal{P}^1), L^*(\mathcal{P}^2)\}|(i^*, j^*)] = Z[L^*]$.

After Theorem 1, we can find the optimal two vertex-disjoint paths by repeatedly removing each edge of the tree T and locating the two optimal paths w.r.t. (13) independently in T_{ij} and in T_{ji}. However, as in [2], this can be done provided that the vertex weights of each subtree are suitably adjusted. Applying the same procedure as in [2], it is possible to obtain a new set of vertex weights, so that problem (13) reduces to finding two independent median paths P_{ij} and P_{ji} in T_{ij} and T_{ji}, respectively.

To conclude this section we summarize the whole procedure to solve 2UMP which consists of the following basic steps.

1. For each vertex r of T

    1.1 transform the given tree into a binary tree rooted at r

    1.2 compute $Z^r(L)$ and find the best pair of paths passing through r and intersecting in the subpath $P(r_1, r_2)$

2. choose the best pair of paths, say (P'_1, P'_2)

3. For each edge (i, j) of T

    3.1 compute the adjusted weights of the vertices in T_{ij} and in T_{ji}

    3.2 find the median paths in T_{ij} and in T_{ji}, and denote them by (P_{ij}, P_{ji}), respectively

4. choose the best (w.r.t. (2)) pair of paths among the pairs (P_{ij}, P_{ji}), for all (i, j), and denote it by $(\bar{P}_{ij}, \bar{P}_{ji})$

5. choose the best solution between (P'_1, P'_2) and $(\bar{P}_{ij}, \bar{P}_{ji})$

Theorem 2 *For a given weighted tree T with n vertices, Problem 2UMP can be solved in $O(n^2)$ time.*

Proof:

Step 1.1 of the above algorithm can be performed in $O(n)$ time by applying the procedure in [9]. The quantities (8), (9) and (10), can be computed in $O(n)$ time by visiting the rooted tree bottom-up, so that finding the best pair (P'_1, P'_2) along with its objective function value takes $O(n^2)$ time. For the second part of the algorithm, after the removal of an edge (i, j), the adjusted vertex weights and the median paths in T_{ij} and T_{ji} can be found in linear time. Thus, also finding the optimal pair of non intersecting paths $(\bar{P}_{ij}, \bar{P}_{ji})$ requires $O(n^2)$ time.

5 Conclusions

We have addressed the problem of locating two path-shaped facilities that minimize the expected service costs in a model where we assume that the paths may become unavailable with disruption probabilities that are known in advance. We showed that the problem in general networks is NP-hard, while we provided a $O(n^2)$ time complexity algorithm for solving it on tree networks. Generalizing to a greater number of facilities, we point out that a similar analysis to the one followed in this paper to solve 2UMP under the assumption that the paths to be located are vertex-disjoint can be applied also to solve 3UMP in $O(n^3)$ time. More generally, the μ Unreliable Median vertex-disjoint Paths with $\mu > 2$ can be solved with the same approach provided that μ is considered as a fixed parameter.

References

[1] Atamturk, A., Zhang, M. (2007). Two-stage robust network flow and design under demand uncertainty. Operations Research 55(4):662–673.

[2] Berman, O., Krass, D., and Menezes, M. (2007). Facility reliability issues in network p-median problems: Strategic centralization and co-location effects. *Operations Research*, 55(2):332–350.

[3] Church, R. L., Scaparra, M. P., and Middleton, R. S. (2004). Identifying critical infrastructure: The median and covering facility interdiction problems. *Annals of the Association of American Geographers*, 94(3):491–502.

[4] Cui, T., Ouyang, Y., and Shen, Z. (2010). Reliable facility location design under the risk of disruptions. *Operations Research*, 58:998–1011.

[5] Drezner, Z. (1987). Heuristic solution methods for two location problems with unreliable facilities. *Journal of the Operations Research Socieety*, 38:509–514.

[6] Drezner, Z. and Hamacher, H. (2004). *Facility Location: Applications and theory*. Springer.

[7] Herivin, H. and Mahjoub, A. (2005). Design of survivable networks: A survey. *Networks*, 46(1):1–21.

[8] Hollander, Y., Prashker, J. (2006). The applicability of non-cooperative game theory in transport analysis. Transportation 33(5):481–496.

[9] Kim T. U., Lowe T. J., Tamir A., Ward J. E. (1996). On the location of a tree-shaped facility. *Networks*, 28:167–175.

[10] Laporte, G., Marn, A., Mesa, J., Perea, F. (2011). Designing robust rapid transit networks with alternative routes. *Journal of advanced transportation* 45:54–65.

[11] Laporte, G., Mesa, J., Perea, F. (2010). A game theoretic framework for the robust railway transit network design problem. *Transportation Research Part B*, 44:447–459.

[12] Liberatore, F., Scaparra, M. P., and Daskin, M. S. (2011). Analysis of facility protection strategies against an uncertain number of attacks: The stochastic r-interdiction median problem with fortification. *Computers & Operations Research*, 38(1):357–366.

[13] Puerto, J., Ricca, F., Scozzari, A. (2009). The continuous and discrete path-variance problems on trees. *Networks*, 53:221–228.

[14] Puerto, J., Ricca, F., Scozzari, A. (2009). Extensive facility location problems on networks with equity measures. *Discrete Applied Mathematics*, 157:1069–1085.

[15] Puerto, J., Ricca, F., Scozzari, A., (2012). Range minimization problems in path-facility location on trees *Discrete Applied Mathematics*, 160:2294–2305.

[16] Puerto, J., Tamir, A. (2005). Locating tree-shaped facilities using the ordered median objective. *Mathematical Programming, Series A*, 102:313–338.

[17] Shen, Z., Shan, R., and Shang, J. (2011). The reliable facility location problem: formulations, heuristics and approximation algorithms. *INFORMS Journal on Computing*, 23(3):470–482.

[18] Snyder, L. and Daskin, M. S. (2005). Reliability models for facility location: The expected failure cost case. *Transportation Science*, 39(3):400–416.

[19] Ukkusuri, S., Mathew, T., Waller, S. (2007). Robust transportation network design under demand uncertainty. Computer-Aided Civil and Infrastructure Engineering 22(1):6–18.

[20] Wood, R. (1993). Deterministic network interdiction. Mathematical and Computer Modeling 17(2):1–18.

[21] Zhang, Y., Berman, O., and Verter, V. (2009). Incorporating congestion in preventive healthcare facility network design. *European Journal of Operational Research*, 198(3):922–935.