



Discrete Optimization

The reliable p -median problem with at-facility serviceMaria Albareda-Sambola^a, Yolanda Hinojosa^b, Justo Puerto^{c,*}^a *Estadística i Investigació Operativa, U. Politècnica de Catalunya, Spain*^b *Economía Aplicada I, U. de Sevilla, Spain*^c *Facultad de Matemáticas, U. de Sevilla, Spain*

ARTICLE INFO

Article history:

Received 18 January 2014

Accepted 30 March 2015

Available online 6 April 2015

Keywords:

Reliable facility location

Integer programming

ABSTRACT

This paper deals with a discrete facility location model where service is provided at the facility sites. It is assumed that facilities can fail and customers do not have information on failures before reaching them. As a consequence, they may need to visit more than one facility, following an optimized search scheme, in order to get service. The goal of the problem is to locate p facilities in order to minimize the expected total travel cost. The paper presents two alternative mathematical programming formulations for this problem and proposes a matheuristic based on a network flow model to provide solutions to it. The computational burden of the presented formulations is tested and compared on a test-bed of instances.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

One of the most important strategic decisions in the design of a supply chain is the location of facilities. This has motivated a lot of research on different facility location problems over the years (see e.g. Drezner & Hamacher, 2004 and references therein) and, in particular, on several discrete facility location problems.

In this context, decision makers may face a lot of uncertainties which can affect not only costs, demands or production capacities, but also facilities availability. That is, in many situations, disruptive events can occur (system failures, natural disasters, terrorist attacks, unexpected work overloads, shortages, labor strikes, etc.) that render facilities temporary unavailable to provide service. These disruptive events can be more or less likely to occur, but when they do, the relocation of the affected customers yields extra costs, which have been taken into account using different strategies.

On the one side, one can focus on identifying robust facility sets that will perform reasonably well even when disruptions occur, either in the underlying network or at the facilities themselves. In this context, performance can reflect customers coverage or distribution costs. Recent examples of robust facility location are Chen, Daskin, Shen, and Uryasev (2006), Kulturel-Konak and Konak (2010), Liu, Guo, Snyder, Lim, and Peng (2010), O'Hanley and Church (2011) or Peng, Snyder, Lim, and Liu (2011). The so-called interdiction models (Aksen & Aras, 2012; Church, Scaparra, & Middleton, 2004;

Liberatore, Scaparra, & Daskin, 2011, 2012, O'Hanley, Scaparra, & García, 2013 etc.) can be seen as the extreme case of robust facility location where failures are not considered to be accidental but the consequence of deliberate attacks. In this case, the most critical failures need to be identified in order to take them into account. A frequent extension to these models is to consider the possibility of fortifying some of the located facilities (see Li, Zeng, & Savachkin, 2013a).

In a real context, no company would accept a supply network with high regular operating costs just to hedge against very rare facility disruptions, unless high penalties are associated with them. Thus, a sensible strategy is to try to balance the regular operating costs and the costs associated with failures. To this end, system configurations should depend on the regular service costs customers and also on the likelihood of the candidate sites to be disrupted. In particular, the so-called *reliable facility location models* aim at minimizing the expected service costs in the long run, assuming that accidental failures occur with probabilities that are known in advance or can be accurately estimated. To the best of our knowledge, reliable facility location was first addressed in Drezner (1987) where both, a median and a center objective function are considered, and the number of facilities that might fail is fixed. These issues were revisited much later in Snyder and Daskin (2005) and other works by the same authors.

Nowadays, reliability issues in supply chain design are of special interest. In particular, studies on system failures due to facility disruptions are gaining attention (Benyoucef, Xie, & Tanonkou, 2013; Morton, Pan, & Saeger, 2007; Pan & Morton, 2008; Qi & Shen, 2007). In the traditional locational analysis literature, Snyder and Daskin (2005) first propose an explicit formulation of the stochastic p -median and fixed-charge problems based on level assignments, where the

* Corresponding author. Tel.: +34 954 557 940; fax: +34 954 622 800.

E-mail addresses: maria.albareda@upc.edu (M. Albareda-Sambola), yhinojos@us.es (Y. Hinojosa), puerto@us.es, justo.puerto@gmail.com (J. Puerto).

candidate sites are subject to random disruptions with equal probability. Alcaraz, Landete, and Monge (2012) propose heuristic algorithms for solving this problem and the works by Zhang, Berman, and Verter (2009) and Berman, Krass, and Menezes (2007b) relax the assumption of uniform failure probabilities. Zhang et al. formulate the stochastic fixed-charged problem as a nonlinear mixed integer program and provide several heuristic solution algorithms. Berman et al. (2007b) focus on an asymptotic property of the problem with a median objective. They show that, when the probability of disruptions is significant, the pattern of optimal locations changes substantially; facility locations become more centralized to provide better support in the event of disruptions. Instead, as the failure probabilities approach zero, facilities tend to spread out to minimize the travel costs, until they coincide with the optimal solution to the problem without failures. Additionally, the authors propose approximation algorithms. More recently, Cui, Ouyang, and Shen (2010) have proposed a Lagrangean relaxation algorithm for this problem as well as a continuous approximation model for it. The same problem has been modeled and approximately solved using a two-stage stochastic program in Shen, Shan, and Shang (2011) and solved exactly using specialized upper and lower bounds in Aboolian, Cui, and Shen (2013). Approximation algorithms have also been developed for this problem (e.g. Guha, Meyerson, & Munagala, 2003), although in this context this family of problems is referred to as fault-tolerant facility location.

Even within this particular context, many problem variants have been considered, depending on the modeling assumptions made. For instance, problems with correlated failures are studied in Berman, Krass, and Menezes (2013) and Li, Ouyang, and Peng (2013b), problems with capacitated facilities in Gade and Pohl (2009) and Aydin and Murat (2013), and the possibility of installing non-failing facilities at a larger cost has also been explored (e.g. Lim, Daskin, Bassamboo, & Chopra, 2010). Other modeling issues concern information availability and customers' behavior upon failures. If the service is at the customers' site, or it is at the facility but customers have complete information about failures, then services can be reassigned to the available facility that is closest to each customer. This assumption is made in most of the existing works. However, there are situations where the service is provided at the facility and the customer cannot know whether a facility is operative or not before reaching it. In these cases, it is reasonable to expect that, after finding their facility unavailable, customers go directly to a new facility from there, instead of traveling back home and choosing their next closest facility for being served. The models considering this assumption are referred to as facility location with incomplete information (see, Berman, Krass, & Menezes, 2009).

Such models allow to tackle many realistic situations. For instance, bank customers travel to their patronized ATM point to withdraw cash and thus they receive their service at the facility location. However, sometimes these machines do not provide service because of network disruptions, maintenance, or simply not servicing some customers' cards and then customers are forced to move from that ATM to a different one, which is likely to be chosen according to its distance to the customer's current location. In some other cases, disruption may be due to simple congestion; three cars in a drive-in window can make it appear as disrupted since no new customer will stop to get service, or long waiting lists in emergency services will imply redirecting ambulances to a different hospital (Witlin (2006) reports that in the US one ambulance is diverted each minute due to emergency room overcrowding), etc. Other examples mentioned in Berman et al. (2009), are gas stations (due to shortage of gas or to long line-ups), retail stores (customers looking to buy a necessary product in a store that is out of stock), or hospital emergency rooms (which regularly experience very long waiting times that may cause newly arriving patients to seek service elsewhere). These situations happen and it is very unlikely that information systems will monitor them in a near future. Next, it is reasonable to assume that, under incomplete information, customers

will usually follow a proximity based *trial-and-error* strategy to visit facilities. They will sequentially try nearby facilities and either stop at the first operative facility or give up after several unsuccessful tries.

Still, under the incomplete information assumption, different customer behaviors can be modeled. Indeed, each time a customer reaches a facility that has failed, she can make her next choice based only on the distance to the next facility or on the expected distance traveled before getting the required service. The first search scheme, used in Berman et al. (2009), responds to common situations where the users do not have the capability to optimize the search for an operating facility. In contrast to this, an optimized search can be assumed for regular services; when someone regularly requires a given service, she gains enough experience to know with quite a good precision the frequency in which the service becomes unavailable (the ATM machine does not work, a very specific product is not available, the taxi stop is empty, the usual parking lot is full . . .). In such cases, it is not rare that a customer uses this knowledge to make the decision to travel in the direction where more services are located or they are more reliable thus trying to reach a facility that on average minimizes total travel time even if it is not the closest one to her current position. Military applications in a war theater of operations can also be modeled assuming an optimized search scheme. This could be the case of units with limited information, due to communication disruption, that try to reach facilities that may have been attacked and are no longer available. In order to improve the survival expectations of their own troops, one action to be developed by high-ranking officers is to design, in advance, contingency plans so that expected travel distances covered by their units is minimized; thus minimizing exposure to further attacks. Another application of this search scheme is in disaster scenarios. In these situations emergency services can, in practice, fail. Moreover, the difficulty to access real-time information is an important issue in critical situations due to communication disruption or facility failure (Galindo & Batta, 2013). In particular, after a large disaster, an emergency center located in the affected area is likely to be destroyed or seriously damaged. Therefore, having contingency plans is important to improve efficiency. This kind of situation was given, for example, in May 11, 2011, in the Spanish town of Lorca: After two consecutive earthquakes nine people were killed, 300 were injured and the local hospital had to be evacuated due to the risk of collapse (Albareda, Hinojosa, Marin, & Puerto, 2015). In all those cases, simply moving to the next closest facility from the one that has failed would make a short-sighted decision.

In this work, we assume that users have incomplete information, and follow an optimized search scheme to find a facility to receive service. In this regard, we have to introduce a routing component in the reliability location model. In order to optimize the system configuration we must keep track of the path potentially followed by each customer until she receives her service. We introduce two different formulations to model the situation described above. Our first formulation uses variables with three indices, including continuous variables that represent the probability that a given arc is traversed by a customer in her path to be serviced. We then present a second formulation in terms of four indices binary variables that trace the paths among facilities using reliability levels. This second formulation is rather intuitive but only applies to the particular case where all the failure probabilities are equal. We compare both formulations and prove that, in this particular case, they are equivalent, the first one providing tighter LP lower bounds. To cope with the very large computing times required to solve our exact formulations we introduce another model that heuristically approximates the Reliability p -Median problem with at-Facility service in its general case. It requires a much lower computing time and provides excellent sets of locations. This model is based on a reduced flow-like formulation that relaxes the assumption that each customer visits a given facility at most once. Computational experiments show that this formulation gives extremely good solutions at a non-expensive computing time.

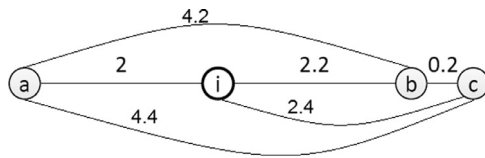


Fig. 1. Example: closer need not be better.

The next section formally describes the Reliability p -Median problem with at-Facility service, discusses some of its properties, and presents a general formulation for it. A reduced formulation, specific for the case with homogeneous failure probabilities is presented in Section 3, and the approximation we propose, in Section 4. Then, in Section 5 we analyze the computational results obtained in a series of instances taken from the literature. Finally, conclusions are outlined in Section 6.

2. The reliability p -median problem with at-facility service

In this section we present the Reliability p -Median problem with at-Facility service (RpMF). The RpMF is an extension of the discrete p -median problem where the possibility that some of the facilities can fail is taken into account. It is assumed that customers travel to a facility for being served and only realize about facility failures on scene. If this happens, they move to other facilities until finding one that is operative. Moreover, each customer $i \in I$ visits the different facilities in the order that minimizes her expected total travel cost, defined as her demand (h_i) times her total expected travel distance without revisiting any facility. The goal in the RpMF is to identify the set of p facilities to open that minimizes the overall expected travel cost.

As in other works on reliable facility location (e.g. Snyder & Daskin, 2005), facilities (J) are divided into two disjoint groups ($J = F \cup NF$); the facilities that can fail (which will happen with a known probability q_j , $j \in F$) and those that will never fail (NF). Moreover, facility failures are assumed to be mutually independent. In addition, the possibility of balking is modeled by means of an extra dummy facility $1 \in NF$, which is forced to be opened. Without loss of generality, we can assume that $I = J$. We will also assume that we are working in a complete connected network, being the length of every arc (j, k) the shortest travel distance between site j and site k . For ease of readability these assumptions will be made in what follows.

In order to describe and evaluate a solution, not only do we need to identify the set of p opened facilities, but we also need to describe the path that each customer will partially follow until getting served. In what follows, we will say that a facility is assigned to a customer at level r if the facility occupies position r in the prospective path of the customer. The reader may observe that for a given subset of facilities assigned to a customer, the optimal assignment levels need not correspond to the order defined by the distances from the customer to these facilities. Moreover, it may be sub-optimal to assign each customer to her closest facility. As the example in Fig. 1 shows, it may be optimal to assign a customer to a facility that is farther away but less likely to fail or closer to other facilities.

In this example, where equal failure probabilities q are assumed, if customer i must travel along the network, the trip obtained following a closest assignment policy is abc . However, depending of the failure probability q (actually, for any $q \in (0.06, 0.89)$) the expected service cost is smaller for the trip bca . This is exactly the trip proposed by our model for these situations.

This policy for defining the customer paths is the difference between the RpMF and the problem studied in Berman et al. (2009), where the closest facility to the current customer location is chosen at each step. While in Berman et al. (2009) it is assumed that, when a facility fails, customers travel to the closest facility to it, in our work,

as in Berman, Ivanovsky, and Krass (2011), it is assumed that the path followed by the customers is the one minimizing their expected total travel time. However, whereas Berman et al. (2011) assumes that the locations of the facilities are fixed, in our model the location of the facilities is a decision to make. At this point, we have to emphasize the essential difference between the model in Berman et al. (2011) and the RpMF: in the former there are no location decisions to make whereas in the latter one has also to decide where to install the facilities. This makes the problem much more difficult since in the RpMF two levels of decision are required, namely, location and routing.

Let d_{jk} be the shortest travel distance from site j to site k , and h_i the demand of customer $i \in I$. Assuming that $I = J$, that we are working in a complete connected network, as mentioned above, and using the following sets of variables,

$$x_j = \begin{cases} 1 & \text{if a facility is opened at location } j \\ 0 & \text{otherwise} \end{cases} \quad j \in I,$$

$$y_{ijk} = \begin{cases} 1 & \text{if arc } (j, k) \text{ is in the path of customer } i \\ 0 & \text{otherwise.} \end{cases} \quad i, j, k \in I$$

and

$w_{ijk} \in [0, 1]$: probability that customer i uses arc (j, k) ; $i, j, k \in I$,

the problem can be formulated as follows:

$$(F1) \min \sum_{i \in I} h_i \sum_{j \in I} \sum_{k \in I} d_{jk} w_{ijk} \quad (1)$$

$$\text{s.t.} \quad \sum_{j \in I} x_j = p \quad (2)$$

$$x_1 = 1 \quad (3)$$

$$\sum_{j \in I} y_{ijk} \leq x_k \quad i, k \in I \quad (4)$$

$$\sum_{k \in I} y_{ijk} \leq 1 \quad i, j \in I \quad (5)$$

$$\sum_{k \in I} w_{ijk} = q_j \sum_{k' \in I} w_{ik'j} \quad i \in I, j \in F, i \neq j \quad (6)$$

$$\sum_{j \in I, j \neq i} w_{ijj} = (1 - x_i) \quad i \in NF \quad (7)$$

$$\sum_{j \in I, j \neq i} w_{ijj} = 1 - (1 - q_i)x_i \quad i \in F \quad (8)$$

$$w_{ijk} \leq y_{ijk} \quad i, j, k \in I \quad (9)$$

$$x_j, y_{ijk} \in \{0, 1\} \quad i, j, k \in I \quad (10)$$

$$w_{ijk} \in [0, 1] \quad i, j, k \in I. \quad (11)$$

Constraints (2) guarantee that exactly p facilities are open, and constraint (3) forces the dummy facility representing the loss of a customer to be one of them. Constraints (4) (resp. (5)) state that each customer can reach (resp. leave) a given facility at most once. Note that, being y variables binary, these constraints would enforce variables x to take binary values and therefore their definition in (10) can be relaxed to be nonnegative. (The reader should observe that a similar relaxation is not possible for the y variables, even keeping integrality for the x variables. Easy examples can be built showing the inconsistency of that relaxation.) Constraints (6) are a kind of transition constraints: if a customer leaves a given facility it means that the facility has failed; thus, the probability that a customer leaves facility j must be q_j times the probability of having reached it. The probability of a given customer leaving its home site is separately set by constraints (7) and (8) for non-failing and failing facilities,

respectively. Finally, constraints (9) force y and w variables to take consistent values.

We observe that in the above formulation, we do not have to force each path to have at most p stops since by constraints (4) it is only allowed to use arcs that arrive at open facilities and each customer can get to any facility at most once. Therefore, altogether implies that there are no cycles and that each customer's tour has at most p stops.

Although the previous formulation is complete, we have considered four intuitive families of additional valid inequalities to strengthen it. The first one relates the probabilities of using given arcs. If arc (j, k) belongs to the prospective path of customer i , the probability that customer i uses it cannot be smaller than the product of the $p - 1$ smallest failure probabilities. Therefore,

$$w_{ijk} \geq \rho_p y_{ijk}, \quad i, j, k \in I, \tag{12}$$

where ρ_p represents the product of the $p - 1$ smallest probability failures among facilities in F . Preliminary computational experiments (shown in Section 5) have proved it useful to add this family of valid inequalities to the above formulation.

In addition, we have considered three more families of valid inequalities. In all cases their number is $O(|I|^2)$ and they can easily be separated by inspection.

On the one hand, we considered the valid inequalities

$$\sum_{j \in I, k \in NF} y_{ijk} \geq 1, \quad i \in F, \tag{13}$$

ensuring that the path followed by any customer i not belonging to NF will always end at a non-failing facility. This inequality has to be satisfied by any feasible solution since in any solution there is, at least, one non-failing open facility (the one modeling the loss of the customer) and any feasible path will end at a non-failing facility ($k \in NF$). On the other hand, analogously to Eqs. (6), flow balance equations on y variables must also hold:

$$\sum_{k \in J} y_{ijk} = \sum_{k' \in I} y_{ik'j}, \quad j \in F, j \neq i. \tag{14}$$

Finally, for $i \in I, j \in F, i \neq j$, Constraints (5) can be reinforced to

$$\sum_{k \in I} y_{ijk} \leq x_j. \tag{15}$$

Preliminary computational tests lead us to solve this formulation including constraints (14) but neither constraints (13) nor (15).

Including constraints (12) and (14), this formulation has $O(|I|^3)$ binary variables, $O(|I|^3)$ continuous variables, and $O(|I|^3)$ constraints.

3. A binary formulation for the homogeneous case

In Formulation F1, continuous variables w are used to apply the right probabilities to the arcs when computing the expected travel costs. These probabilities are, in fact, products of failure probabilities of different sets of facilities. Consequently, in the particular case where the failure probabilities are homogeneous, all those probabilities are powers of the common failure probability. Therefore, in this case, they can be alternatively included in the formulation by considering an extra index in the y variables, to indicate the customer assignment levels. This is inspired in the ideas used in Berman et al. (2009), when modeling a similar problem, but with service at the customers' site (and, thus, assignments to the closest facilities) and in Snyder and Daskin (2005) where different assignment levels are used to represent the order in which the open facilities will serve a customer in case of failures.

To implement this idea, we will use again variables:

$$x_j = \begin{cases} 1 & \text{if a facility is opened at location } j \\ 0 & \text{otherwise} \end{cases} \quad j \in I,$$

now together with:

$$y_{ijkr} = \begin{cases} 1 & \text{if customer } i \text{ travels from facility } j \text{ to} \\ & \text{facility } k \text{ at level } r \quad i, j, k \in I, r \in \{1, \dots, p\}. \\ 0 & \text{otherwise,} \end{cases}$$

Taking into account that we have assumed $I = J$, variables y_{ijkr} are not defined (or, equivalently, fixed to zero) in the following cases:

- y_{ijjr} , for any $i, j \in I, r \in \{1, \dots, p\}$ since no customer will be assigned to the same facility in two different levels.
- y_{iikr} , for $r > 2$ and any $i, j \in I$, since a customer must travel from its origin to a facility before level 2.
- y_{ijir} , for any $i, j \in I, r \in \{1, \dots, p\}$, since it makes no sense that a customer goes back to its home place after visiting another place.

Using the above variables and assuming that all the failure probabilities are equal ($q_j = q, j \in J$), the RpMF can be modeled as follows:

$$(F2) \quad \min \quad \sum_{i \in I} h_i \sum_{r=1}^p q^{r-1} \sum_{j \in I} \sum_{k \in I} d_{jk} y_{ijkr} \tag{16}$$

$$\text{s.t.} \quad \sum_{j \in I} x_j = p \tag{17}$$

$$x_1 = 1 \tag{18}$$

$$\sum_{j \in I} y_{ijkr} = \sum_{\ell \in I} y_{i\ell r+1} \quad i \in I, r = 2, \dots, p - 1, k \in F \tag{19}$$

$$\sum_{j \in I} y_{ijk1} \leq \sum_{\ell \in I} y_{i\ell k2} \quad i \in I, k \in F \tag{20}$$

$$\sum_{r=1}^p \sum_{k \in I} y_{ijkr} \leq 1 \quad i, j \in I \tag{21}$$

$$\sum_{r=1}^p \sum_{j \in I} y_{ijkr} \leq x_k \quad i, k \in I \tag{22}$$

$$\sum_{k \in I} y_{iik1} + x_i = 1 \quad i \in I \tag{23}$$

$$\sum_{k \in I} y_{iik2} = x_i \quad i \in F \tag{24}$$

$$y_{ijkr} \leq (1 - x_i) \quad i \in NF, j, k \in I; r = 1, \dots, p \tag{25}$$

$$x_j, y_{ijkr} \in \{0, 1\} \quad i, j, k \in I; r = 1, \dots, p. \tag{26}$$

Now, the objective function (16), computes the sum of the expected service costs for all the customers. Equation (17) fixes the number of facilities to establish, and Eq. (18) forces the dummy facility used to model balking to be among them. Constraints (19) and (20) ensure that if a customer goes to a failing facility at a given level $r < p$ it leaves that facility in the next level. To make sure that a facility will not be used in more than one level by the same customer, we use constraints (21). In the same way, we use (22) to make sure that a customer can only use a facility at some level if it is open. Equations (23) are used to ensure that, for a given customer, either it becomes a facility, or it has a primary assignment. Analogously, we use constraints (24) to ensure that if a failing facility is established at a given point, the corresponding customer will have a secondary assignment. In the case of non-failing facilities, constraints (25) guarantee that if a non-failing facility is settled, its corresponding customer will not be assigned to any other facility. Finally, we easily observe that we can relax x variables to take continuous values rather than binary ones since constraints (22) will enforce this property from the y variables.

We observe that formulation F2 uses essentially the same set of variables as the formulation presented by Berman et al. (2009). However, the variations with respect to that formulation are essential to

reflect the different modeling assumptions: the different behavior of the customers looking for at-facility service and the existence of non-failing facilities (as a physical reality, or just to model situations where a customer is lost).

Note that this formulation has $O(p|I|^3)$ binary variables and $O(p|I|^2)$ constraints. The main reason for using so many variables is the need of keeping track of the path followed by each customer, to ensure that no customer will visit the same facility twice. This formulation is as intuitive as F1 but it does not mix binary and continuous variables. On the other hand it uses a larger number of variables, and is only valid for the particular case where all failure probabilities are equal. For this case, we show next that the two formulations are equivalent.

Proposition 3.1. *For the particular case with homogeneous failure probabilities, formulations F1 and F2 are equivalent.*

Proof. Consider an instance of the RpMF where all the failure probabilities are equal ($q_j = q \forall j \in J$). Let (x, y) be a feasible solution of Formulation F2. We will see that there exists a feasible solution of Formulation F1 $(\bar{x}, \bar{y}, \bar{w})$ such that the objective function values (16) and (1) are equal. Conversely, from any feasible solution of Formulation F1 $(\bar{x}, \bar{y}, \bar{w})$ we will derive a feasible solution of Formulation F2 (x, y) such that their objective function values coincide.

- Assume that (x, y) is a feasible solution of Formulation F2. Then $\bar{x}_j = x_j$, $\bar{y}_{ijk} = \sum_{r=1}^p y_{ijk_r}$ and $\bar{w}_{ijk} = \sum_{r=1}^p q^{r-1} y_{ijk_r}$ for all i, j, k , defines a feasible solution of F1 with the same objective value as (x, y) in F2.
- Conversely, let $(\bar{x}, \bar{y}, \bar{w})$ be a feasible solution of Formulation F1. Observe that, thanks to constraints (5) and (9), for a given pair i, j , \bar{w}_{ijk} can only be different from zero for one k index. Together with this condition, constraints (6)–(8) force \bar{w} variables to take values that are powers of q . Thus, equations $\sum_{r=1}^p q^{r-1} y_{ijk_r} = \bar{y}_{ijk} \bar{w}_{ijk}$ for all i, j, k , have a binary solution y_{ijk_r} . This solution together with $x_j = \bar{x}_j$ for all $j \in I$ define a feasible solution of F2 with the same objective value as $(\bar{x}, \bar{y}, \bar{w})$ in F1. \square

Although this new formulation is larger than F1 in terms of number of variables, it has a smaller number of constraints, which in F1 is dominated by the $O(|I|^3)$ sets of constraints linking y and w variables. As it will be seen later in the computational experience, solving F2 to optimality using standard MIP solvers is computationally less demanding than solving F1. Nevertheless, it still has strong computational requirements. Moreover, it can only be used for the homogeneous case. For this reason, in the next section we present a relaxation of the RpMF that is valid for the general case, whose formulation can be solved with a smaller computational effort. We evaluate the heuristic solutions to the RpMF obtained by solving this relaxation in Section 5.

4. Flow approximation to the p -reliable facility location problem with at-facility service

In the RpMF it is assumed that a given customer visits each of the p opened facilities at most once. In this section we consider the problem that is obtained by relaxing this assumption. Without this assumption, it is no longer needed to keep track of the path followed by a given customer, since now customers are allowed to revisit facilities. A natural consequence of relaxing this assumption is the following: if a given facility $j \in I$ fails, all customers that have reached that location, independently of the path they have followed before, will continue their path to the same next facility j' . In what follows, we will refer to j' as the backup facility of j .

The program presented in this section takes advantage of this fact to formulate the problem using the following variables:

- $x_j = \begin{cases} 1 & \text{if facility } j \text{ is opened} \\ 0 & \text{otherwise,} \end{cases} \quad j \in I.$

- $y_{j'j} = \begin{cases} 1 & \text{if the primary assignment of point } j \text{ is to } j' \\ 0 & \text{otherwise,} \end{cases} \quad j, j' \in I.$
- $w_{j'j} =$ expected total flow from j to j' , $j, j' \in I.$
- $u_{j'j} = \begin{cases} 1 & \text{a facility is opened at site } j \text{ and } j' \text{ is its backup facility} \\ 0 & \text{otherwise,} \end{cases} \quad j, j' \in I.$

A formulation of the proposed relaxation for the general case of the RpMF follows.

$$(FP) \min \sum_{i \in I} \sum_{j \in I} d_{ij} w_{ij} \tag{27}$$

$$\text{s.t.} \quad \sum_{j \in I} x_j = p \tag{28}$$

$$x_1 = 1 \tag{29}$$

$$\sum_{k \in I} w_{jk} = q_j \left(\sum_{i \in I} w_{ij} + h_j x_j \right) + h_j (1 - x_j) \quad j \in F \tag{30}$$

$$\sum_{k \in I} w_{jk} = h_j (1 - x_j) \quad j \in NF \tag{31}$$

$$w_{jk} \geq h_j y_{jk} \quad j \neq k \in I \tag{32}$$

$$w_{jk} \leq M u_{jk} + h_j y_{jk} \quad j \neq k \in I \tag{33}$$

$$\sum_{k \in I} u_{jk} = x_j \quad j \in F \tag{34}$$

$$x_j + \sum_{k \in I, k \neq j} y_{jk} = 1 \quad j \in I \tag{35}$$

$$u_{jk} \leq x_k \quad j \in F, k \in I \tag{36}$$

$$y_{jk} \leq x_k \quad j, k \in I \tag{37}$$

$$x_j, u_{jk}, y_{jk} \in \{0, 1\} \quad j \neq k \in I \tag{38}$$

$$w_{jk} \geq 0 \quad j \neq k \in I. \tag{39}$$

Constraints (28) and (29) are equivalent to (17) and (18) in formulation F2 of the RpMF; they set the number of facilities to open and force the dummy facility representing the loss of a customer being one of them. Constraints (30) are flow conservation constraints at nodes in F . If no facility is settled at location $j \in F$, the whole demand h_j must leave node j , otherwise, the expected flow leaving node j is equal to the probability that j fails times the flow at j , which is equal to the entering flow, plus the demand at j . Analogous constraints for nodes in NF are given by Eqs. (31). Next, constraints (32) force primary assignments to be consistent. If facility $k \neq j$ is the primary assignment of customer j , then all the demand of customer j must go through arc (j, k) . On the other hand, constraints (33) force u and y variables to be consistent with flows given by w , namely, if some flow goes from node j to k it is because some customer travels between these two facilities. Here, M must be an upper bound on the expected flow from node j to node k , in solutions where a facility is located at j . The value that has been used for this bound in the computational experiments is $q_j \sum_{i \in I} h_i$. Next, constraints (34) link u and x variables, making sure that any open failing facility has exactly one backup facility assigned. Finally, constraints (35) ensure that each node has either an open facility or a primary assignment. The remaining constraints are logical links between the binary variables. We observe once more that x variables might be defined as continuous since (37) will ensure their binary character.

This program involves $O(|I|^2)$ variables and $O(|I|^2)$ constraints. Therefore, it is much smaller than any of the proposed formulations for the RpMF.

Table 1
Number of instances.

	Original dataset	Sampled sizes n	Number of facilities p	Number of replications per (n, p)	Total number of instances
Type I instances	49 nodes	20, 25, 30, 35, 40, 45	4, 5, 6	3	54
	88 nodes	20, 25, 30, 35, 40, 45	4, 5, 6	3	54
	150 nodes	20, 25, 30, 35, 40, 45	4, 5, 6	3	54
Total					162
Type II instances	50 nodes	20, 25, 30, 35, 40, 45	4, 5, 6	3	54
	100 nodes	20, 25, 30, 35, 40, 45	4, 5, 6	3	54
Total					108
Total					270

Note that the feasible region of FP strictly contains that of the RpMF since paths that are not feasible for the RpMF are considered in FP. Indeed, if the expected length of a potentially infinite loop among opened facilities is not longer than the length of an elementary path leading to a non-failing facility, the obtained solution can contain such a loop, yielding an underestimate of the real expected service cost of a customer. Therefore, the optimal FP value is always a valid lower bound to the optimal value of the RpMF. Observe also that the x -parts of the feasible solutions of FP provide partial solutions for the RpMF that can be completed in a straightforward way: once the open facilities are fixed, the optimal paths associated with each customer can be easily found by fixing the location variables in formulation F1. Thus, solving formulation FP we can obtain both, a feasible solution to the RpMF, and a lower bound to its optimal value.

We have carried out a series of computational experiences to assess the quality of the sets of facilities given by this approximation as solutions for the RpMF, and to evaluate the computational effort required to find them. The obtained results are reported in the following section. Finally, it is worth mentioning that the approach presented in this section is close to the one presented in [Berman et al. \(2009\)](#). In that paper the authors bring up the idea of common paths in a different but related way. Also [Berman, Huang, Kim, and Menezes \(2007a\)](#) present a somehow similar approach, although in a queueing context.

5. Computational experience

5.1. Comparison of all formulations: homogeneous instances

To evaluate the difficulty of solving the different formulations presented in this work and the quality of the heuristic solutions to the RpMF given by formulation FP we have generated a series of instances that are described next.

Note that the set of data required to define a problem instance with at-facility service is the same as for the problem with at-customer service, except that in the former fixed costs for opening facilities are not considered. As a basis for generating our instances, we took the instances used in [Snyder and Daskin \(2005\)](#) for the problem with at-customer service. In that work, five instances were considered, divided into two groups. The first group contains three instances of 49, 88, and 150 nodes, respectively. These nodes are taken as cities in the United States with demands proportional to the city population, distances equal to the great circle distances among them, and a cost for missing a customer equal to 10^4 . In those instances, the only non-failing facility is the dummy facility that models losing a customer. The second group contains two randomly generated instances with 50 and 100 nodes taken from $[0, 1] \times [0, 1]$. In that set, Euclidean distances are considered, demands are taken in $[0, 1000]$, and the cost for missing a customer is set to 10. In all cases, to be able to compare the three formulations, a common failure probability is used for all the facilities that belong to F , which is taken to be $q = 0.05$. The dummy facility representing the loss of a customer is not taken into account in the dimensions (n, p) of the instance, and thus, $n = |I| - 1$. Note that the possible applications of the RpMF are mostly

defined in quite local settings. Therefore, the instances defined on different sets of North-American cities may look somehow inappropriate. However, we think the obtained results would be the same with distances within a smaller scale, and using the same data as in [Snyder and Daskin \(2005\)](#) allows to compare the computational burden of the problem variants with at-facility and at-customer service. In what follows we will refer the first group of instances as Type I instances, and the second one as Type II instances.

In order to generate smaller instances, we randomly selected subsets of nodes from the original data sets with sizes $n \in \{20, 25, 30, 35, 40, 45\}$ and considered values of $p \in \{4, 5, 6\}$. In particular, for each original instance and each dimension (n, p) we generated three different new instances, giving rise to a total of 270 instances. This information is summarized in [Table 1](#).

The three formulations presented in this work were implemented using C and CPLEX 12.1 callable libraries and a time limit of 1 hour of CPU time was set. All the experiments were run on a personal computer running under Windows XP equipped with an Intel Core Duo E8500 processor at 3.16 gigahertz with 3.4 gigabyte of RAM.

The CPU times required for solving the RpMF on the generated instances using Formulation F1 were extremely large; they exceeded the CPU time limit of 1 hour in 56 of the 162 Type I instances, and in 64 of the 108 Type II instances.

In the case of formulation F2 the CPU times, in seconds, are depicted in [Fig. 2](#). In these figures we have considered separately Type I and II instances since, as can be seen, their behavior is quite different. This fact can be attributed to the difference in the relationship between node distances and failure probabilities among these sets, and also to the fact that Type I instances are based on real data which results in non-uniform distributions of the demands while Type II instances are based on random (uniform) distributions of demands. For each instance class (Types I and II) and size (n, p) the average CPU time is plotted. Note that these averages consider either nine or six instances, depending on the class. In both cases, however, the value of the number of medians to settle does not show a great impact on the computational burden of formulation F2. Conversely, both, the number of customers and the instance class seem to have a strong effect on the effort required to solve the RpMF using formulation F2. In both classes the difficulty of the instances increase fast as they become larger, but they are consistently more difficult in the Type II class, where the penalty for losing a customer is smaller, as compared to the distances between the nodes and, thus, the dummy non-failing facility is more attractive.

Despite the fast growth of the computational effort required to solve the considered instances, no instance in the Type I class required more than 3 minutes, and all instances in the Type II class could be solved in, at most, 10 minutes. These encouraging results lead us to try to solve larger instances. To this end, we considered the five full instances from [Snyder and Daskin \(2005\)](#) (with n ranging from 49 to 150) combined with $p \in \{5, 10, 20\}$. For these instances, we set a CPU time limit of 2 hours. In the case of formulation F2, CPLEX was not even able to load the problem in most cases. In contrast, FP was able to solve 10 out of the 15 instances in less than 2 hours, including the

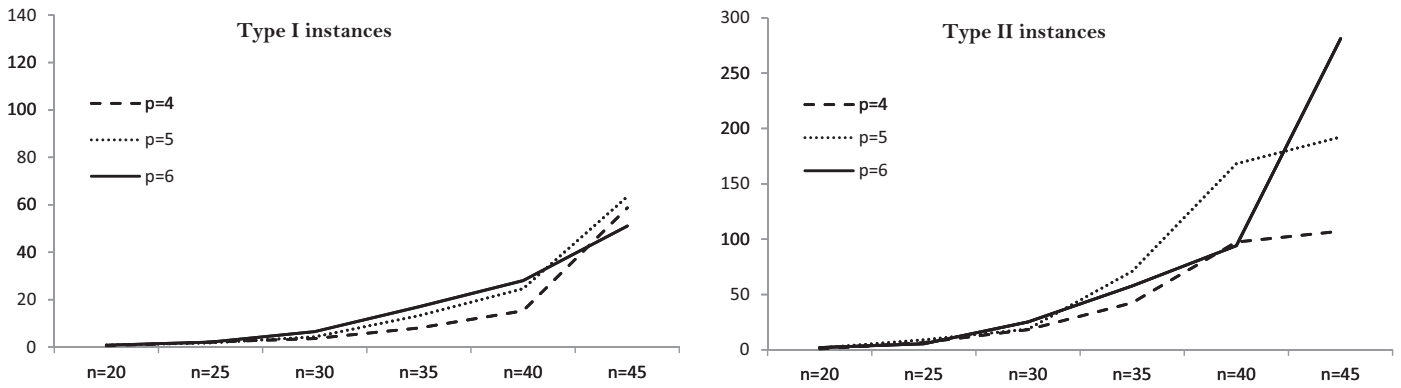


Fig. 2. CPU times (seconds) for solving formulation F2.

Table 2
FP: Percent gaps at termination for unsolved instances.

Instance	n	p	Percent gap
88UFLP	88	5	5
88UFLP	88	10	3
100EuclUFLP	100	5	7
100EuclUFLP	100	10	8
100EuclUFLP	100	20	5

Type I instances with $n = 150$. Indeed, none of them took more than 1 hour, and the average time was 743.5 seconds. Table 2 gives the percent gaps at termination for the five remaining instances.

We next analyze the lower bounds obtained with the LP relaxations of formulations F1, F2 and FP, as well as the lower bound given by the optimal value of formulation FP. Tables 3 and 4 report the percent gaps from the optimal solution associated with them. Average values among the three instances of the same dimension generated from a given original instance are given. Again, results for Types I and II instances are reported separately.

In both tables, under headings n and p we give the dimension of the instances, without taking into account the dummy facility used to model the penalty for not serving a customer. Under the heading origin we give the name of the source instance, the gaps corresponding to the LP relaxations of F1, F2 and FP are given under the headings LF1, LF2 and LFP, respectively. Finally, under heading FP we give the percent gaps of the lower bound given by the optimal value of program FP.

As can be seen in Tables 3 and 4, formulation F2 is rather loose and the LP gap it provides is the largest, specially for the smaller instances and large values of p in both, Types I and II instances. The smallest LP gap obtained with this formulation was over 15 percent. The average gap for instances with four medians was 33 percent, for instances with five medians it was 41 percent and for instances with six medians, 48 percent. Nevertheless, the average CPU time required to find this bound for Type I instances was 5.9 seconds, and for Type II instances, 7.5.

Tables 3 and 4 also show that formulation F1 is much stronger than formulation F2, in terms of the quality of its LP lower bound. Indeed, the percent deviation of this lower bound with respect to the optimal value was, on the average, 7.5 percent. Moreover, it never exceeded 22 percent and was under 10 percent in 214 of the 270 instances. In the case of formulation F1, the value of p does not seem to have any effect on the quality of the LP bound. Also, similar gaps are observed in Types I and II instances. As in the case of formulation F2, percent gaps seem to be slightly smaller for larger instances. Times required to compute this lower bound are similar to the ones obtained with formulation F2.

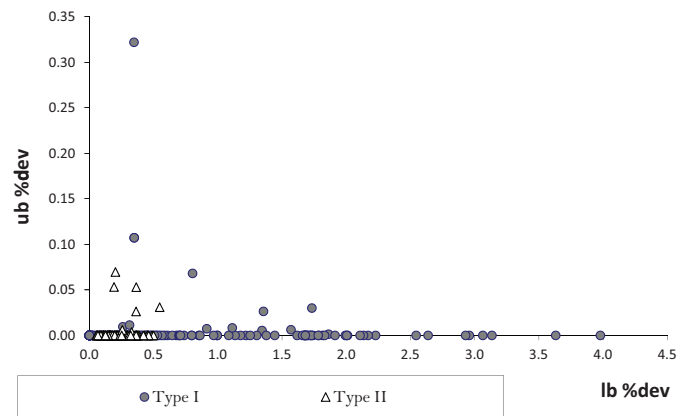


Fig. 3. Percent deviations of the FP bounds.

Surprisingly, the lower bound given by the LP relaxation of the approximation of the RpMF given by FP is only slightly worse than the LP bound of formulation F1 and much better than the LP bound of F2. On the average, the percent gaps given under LF2 are four times larger than the ones given under LFP and those, are twice the ones under LF1. For 109 of the 270 instances this gap was under 10 percent and, they ranged in (0.8 percent, 28.9 percent).

As expected, the lower bound given by the optimal solution of FP is very tight. Indeed, the average gap over the 270 instances was 0.54 percent and it never exceeded 4 percent. Moreover, for 34 instances it was exactly 0, and for 222 it was under 1 percent. However, as we will report later, obtaining these tight bounds required larger CPU times.

To better illustrate the quality of the solutions given by the approximation FP, Fig. 3 shows the percent deviations of the upper and the lower bounds obtained with FP from the optimal value, for each instance. As can be seen in this figure, the results obtained with this approximation are very good, specially in the case of Type II instances, where no solution was more than 0.1 percent away from the optimal, and the lower bounds never reached a deviation of 1 percent.

The CPU times required to solve FP for the different instances are shown in Fig. 4 exactly in the same way they were reported for formulation F2 in Fig. 2. As this figure shows, similar patterns to those observed for formulation F2 can be seen, although differences are now smoother; the difference between Types I and II instances is smaller, and the growth of time requirements as the number of customers increases is quite slower.

Next, we analyze the quality of the heuristic solutions for the RpMF built from the solutions of FP. The percent deviation of the values of such solutions with respect to the optimal solution was, on the average, 0.002 percent. It never exceeded 0.35 percent and it was

Table 3
Lower bounds: Percent gaps from the optimum for Type I instances.

<i>n</i>	<i>p</i>	Origin	LF1	LF2	LFP	FP
20	4	150UFLP	5.54	41.10	10.12	0.23
		49UFLP	6.59	47.80	12.19	2.08
		88UFLP	9.67	54.23	20.18	3.24
	5	150UFLP	6.77	54.76	11.23	0.10
		49UFLP	7.87	59.17	11.52	0.44
		88UFLP	13.69	63.83	17.63	0.88
	6	150UFLP	6.22	64.68	11.56	0.30
		49UFLP	7.29	69.89	11.61	0.47
		88UFLP	12.59	71.80	17.67	1.20
25	4	150UFLP	2.85	29.00	7.69	0.39
		49UFLP	9.72	42.70	15.00	1.91
		88UFLP	14.71	49.15	24.83	3.18
	5	150UFLP	4.33	40.42	8.28	0.09
		49UFLP	8.71	47.85	13.04	0.36
		88UFLP	13.92	54.50	17.50	0.92
	6	150UFLP	5.11	49.59	9.66	0.39
		49UFLP	10.61	58.61	14.10	0.32
		88UFLP	10.03	61.40	14.18	1.30
30	4	150UFLP	3.36	23.45	6.89	0.42
		49UFLP	9.46	35.20	14.76	1.80
		88UFLP	13.24	42.02	21.25	2.46
	5	150UFLP	3.57	31.52	6.83	0.00
		49UFLP	9.38	40.42	13.07	0.31
		88UFLP	12.25	48.37	16.82	0.62
	6	150UFLP	3.65	39.53	7.11	0.33
		49UFLP	9.93	48.26	13.25	0.42
		88UFLP	10.25	54.72	13.94	0.84
35	4	150UFLP	4.02	26.68	9.25	0.00
		49UFLP	7.67	35.57	13.07	1.72
		88UFLP	13.44	34.03	17.75	1.84
	5	150UFLP	6.72	36.00	10.42	0.00
		49UFLP	7.96	39.85	11.49	0.21
		88UFLP	11.56	39.23	15.61	0.49
	6	150UFLP	7.06	44.72	11.01	0.07
		49UFLP	7.61	45.12	10.99	0.26
		88UFLP	9.46	45.47	13.38	0.62
40	4	150UFLP	3.90	21.79	7.73	0.00
		49UFLP	7.56	31.84	12.28	1.69
		88UFLP	13.65	32.06	18.99	1.85
	5	150UFLP	3.65	29.40	5.68	0.01
		49UFLP	7.88	36.36	11.29	0.28
		88UFLP	12.42	37.22	16.06	0.56
	6	150UFLP	2.77	35.17	5.94	0.03
		49UFLP	7.50	40.19	10.94	0.18
		88UFLP	10.37	41.56	14.43	0.56
45	4	150UFLP	2.99	18.48	4.68	0.00
		49UFLP	7.92	30.66	12.17	1.59
		88UFLP	12.25	27.81	17.57	1.79
	5	150UFLP	2.07	24.89	3.88	0.09
		49UFLP	7.85	33.77	11.30	0.25
		88UFLP	11.23	32.83	14.45	0.59
	6	150UFLP	2.38	31.53	4.53	0.06
		49UFLP	7.96	37.33	11.23	0.10
		88UFLP	9.48	37.22	13.78	0.49

0 in 251 of the 270 instances. For the instances where this heuristic solution was not optimal we compared the set of opened facilities in this solution with the same set in the optimal solution found solving F2. In all of these instances the sets contained one single different facility except for two instances (both with $p = 5$) where the difference was in two facilities.

Summarizing, formulation F2 has strong memory requirements but, despite having quite loose LP bounds can be solved within reasonable amounts of time when the memory requirements are affordable. Formulation F1 is more general than F2, since it can cope with

Table 4
Lower bounds: Percent gaps from the optimum for Type II instances.

<i>n</i>	<i>p</i>	Origin	LF1	LF2	LFP	FP
20	4	100EuclUFLP	5.67	40.92	10.48	0.41
		50EucUFLP	5.15	46.87	8.50	0.35
	5	100EuclUFLP	7.79	54.68	11.94	0.22
		50EucUFLP	5.21	58.25	8.50	0.24
	6	100EuclUFLP	6.96	66.25	10.82	0.24
		50EucUFLP	5.49	66.50	9.46	0.19
25	4	100EuclUFLP	6.32	34.28	9.93	0.29
		50EucUFLP	5.95	41.21	9.03	0.32
	5	100EuclUFLP	6.41	44.63	9.92	0.20
		50EucUFLP	6.05	50.52	10.06	0.22
	6	100EuclUFLP	6.70	52.87	11.45	0.23
		50EucUFLP	5.97	58.05	10.56	0.20
30	4	100EuclUFLP	6.46	28.19	9.96	0.23
		50EucUFLP	5.27	34.54	8.25	0.30
	5	100EuclUFLP	6.88	37.39	9.87	0.27
		50EucUFLP	5.94	43.57	9.64	0.27
	6	100EuclUFLP	6.68	45.20	10.22	0.23
		50EucUFLP	6.04	49.95	10.51	0.17
35	4	100EuclUFLP	7.04	26.47	10.23	0.18
		50EucUFLP	6.58	27.60	9.45	0.32
	5	100EuclUFLP	7.48	35.84	10.87	0.20
		50EucUFLP	5.76	36.13	8.87	0.22
	6	100EuclUFLP	6.11	43.24	9.95	0.27
		50EucUFLP	5.87	43.80	9.79	0.15
40	4	100EuclUFLP	7.34	23.07	10.47	0.17
		50EucUFLP	7.53	23.63	10.29	0.36
	5	100EuclUFLP	7.27	29.21	11.25	0.21
		50EucUFLP	7.28	31.88	10.27	0.17
	6	100EuclUFLP	7.23	35.99	11.26	0.22
		50EucUFLP	6.82	38.69	10.68	0.14
45	4	100EuclUFLP	7.23	20.00	10.84	0.13
		50EucUFLP	6.98	19.93	9.57	0.22
	5	100EuclUFLP	7.47	26.38	10.87	0.13
		50EucUFLP	6.44	27.36	8.90	0.25
	6	100EuclUFLP	7.41	32.99	10.89	0.18
		50EucUFLP	6.61	34.56	10.06	0.11

situations with non-uniform failure probabilities. This formulation is more compact than F2 and this reduces its space requirements, which allows to solve larger instances. However, the time requirements for solving them using standard solvers is very large. This is probably due to the fact that it is a mixed integer programming formulation and in the constraints linking binary and continuous variables there are some coefficients that can be very small as compared to the coefficients in the other constraints. This causes this formulation to be ill-conditioned and increases its computational burden. Finally, using the approximation given by FP we obtain both, heuristic solutions and lower bounds for the RpMF. The space requirements of this formulation are much smaller than those of F1, and it can be solved in times that are, on the average, smaller than the times needed to solve F2. The obtained solutions were optimal in most cases, and very close to optimal otherwise. Moreover, this approximation can be used also in situations where failure probabilities are different.

Additionally to the above experiments, we have solved the p -median problem (pM) and the reliability p -median problem with at-customer service ($RpMC$) on these homogeneous instances, and we have evaluated their optimal solutions (sets of facilities) as solutions to the $RpMF$.

In 91 instances, the set of facilities in the pM solution was optimal for the $RpMF$ problem with $q = 0.05$, but it was far from optimal in the remaining ones. The average deviation of the corresponding solution

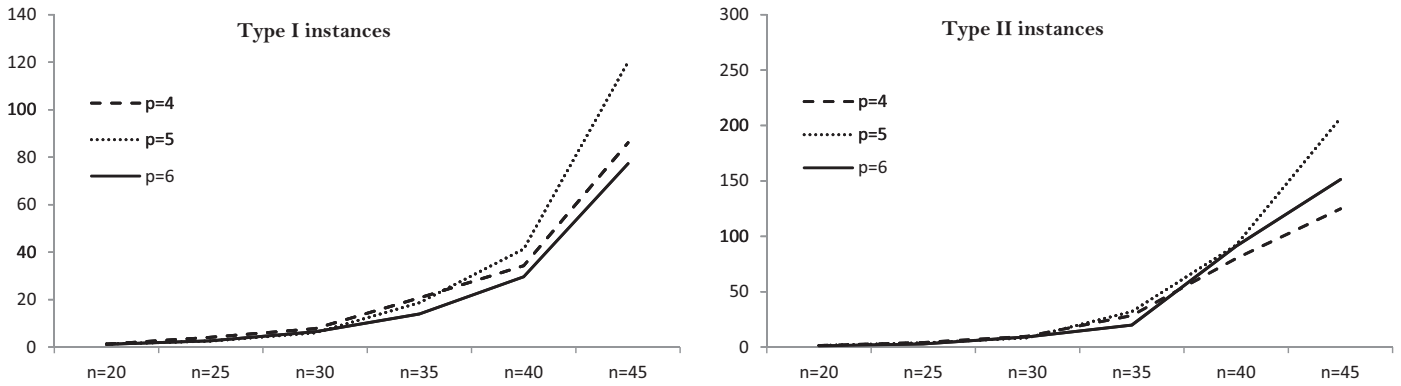
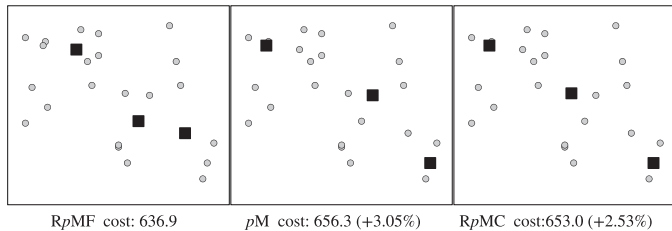


Fig. 4. CPU times (seconds) for solving FP.



RpMF cost: 636.9 pM cost: 656.3 (+3.05%) RpMC cost: 653.0 (+2.53%)

Fig. 5. Example: Comparison of the solutions to RpMF, pM and RpMC.

from the optimal RpMF solution was 2.89 percent and it reached 24.65 percent in one instance. In the case of the RpMC instances, the set of facilities in their optimal solution was optimal for the RpMF in 167 instances but, on the average, it yielded a RpMF solution with a deviation of 0.13 percent from the optimum. According to these results, at least for small failure probabilities, the RpMC and the RpMF have similar solutions although they need not be the same. As opposite, pM solutions can be quite weak as solutions for the RpMF. A detailed example of this fact is given in Fig. 5. In this example, a set of 25 customers with one unit of demand each have been randomly generated in the 100 × 100 square, and the corresponding RpMF instance has been generated considering Euclidean distances, a penalty for losing a customer equal to 200, a failure probability $q = 0.05$, and $p = 3$. The figure shows the optimal sets of facilities for the RpMF, the pM and the RpMC, respectively, their associated cost as solutions to the RpMF and the corresponding percent deviation from the optimal solution. As it can be seen, beyond the difference in the solution costs, the three sets of facilities are quite different. Indeed, the facility concentration effect already observed by other authors in the case of the RpMC, is even stronger in the case of the RpMF.

Summarizing, we can conclude that it is important both, to include failure probabilities into consideration when facilities are indeed prone to failure, and to model adequately the actual service policies. Otherwise, excessively simplified models will yield solutions that can be quite far from optimal. From the above analysis users may decide, based on their own requirements, when to apply the more precise RpMF model, at the price of higher computation burden, or an approximated model, RpMC, still giving reasonably good solutions. In any case, one should not forget that the RpMF is a new available model that adds another entry in the reduced list of tools for handling facility failures or disruption.

5.2. General instances

All the computational experiments carried out in the previous section were performed on instances with homogeneous failure probabilities. We next report the results obtained in a second series of

Table 5

Average of CPU times (s) for instances with site-dependent failure probabilities.

	Formulation F1		FP approx.	
	n = 20	n = 25	n = 20	n = 25
$q \in 0.03 \pm 0.003$	7.39	48.00	0.56	1.50
$q \in 0.03 \pm 0.025$	4.78	29.49	0.53	1.42
$q \in 0.07 \pm 0.003$	112.19	1255.88	0.75	2.30
$q \in 0.07 \pm 0.025$	122.92	436.06	0.83	2.24
$q = 0.05$	20.58	102.36	1.36	4.03

experiments carried out on instances with site-dependent failure probabilities. Of course, for this second set of experiments, only formulation F1 and the approximation given by FP are used.

Instances with site-dependent failure probabilities were generated from the 30 instances with $n \in \{20, 25\}$ and $p = 4$ of the previous section. Probabilities were randomly taken from 4 different intervals $m \pm s$ with $m \in \{0.03, 0.07\}$ and $s \in \{0.003, 0.025\}$ yielding a total of $30 \times 4 = 120$ instances.

Using formulation F1 we could solve 55 instances with $n = 20$ and 45 with $n = 25$ within 2 hours of CPU time. Branch and bound was stopped because of the CPU time limit in 33 percent of the remaining instances, and because of memory shortage in the rest. All these cases correspond to instances with $m = 0.07$. When using the approximation given by FP, all instances could be solved in less than 15 seconds. The medians of the required CPU times are reported in Table 5. For ease of comparison, the table also contains the medians CPU times for the instances with homogeneous failure probabilities of the same dimensions. As can be seen, the effect of the value and the variability of the failure probabilities is very different in the case of F1 and FP. As the variability of the failure probabilities increases, instances tend to become a bit easier to solve, specially in the case of formulation F1. Moreover, the increase of these probabilities has a dramatic effect on the computational burden of formulation F1. Indeed, as mentioned above, even for small sizes as the ones considered in this experiment, some instances could not be solved to optimality within 2 hours of CPU time. Even though FP also becomes harder to solve as the probabilities increase, the effect is much softer in this case.

In concordance with what was observed in the homogeneous case, the solutions provided by the FP approximation were always very good. In the case of the 60 instances with $n = 20$ there was one single instance for which the solution did not coincide with the one provided by F1 (regardless the termination criterion of F1), and its deviation from optimal was 0.13 percent. In the case of instances with $n = 25$, the solution provided by FP coincided with the one obtained with F1 in 52 instances, it was better in six instances where F1 terminated

Table 6
Number of instances (out of 15) with coincident solutions (set of facilities).

	n = 20				n = 25			
	A	B	C	D	A	B	C	D
A: $q \in 0.03 \pm 0.003$	–	10	10	8	–	6	6	3
B: $q \in 0.03 \pm 0.025$	10	–	8	10	6	–	2	9
C: $q \in 0.07 \pm 0.003$	10	8	–	7	6	2	–	5
D: $q \in 0.07 \pm 0.025$	8	10	7	–	3	9	5	–

because of the limits on CPU time or memory, with deviations down to – 5 percent and it was worse in two instances, with a maximum deviation of 1.41 percent.

To conclude this experiment, we gathered the solutions obtained with formulation F1 for all instances, and intervals of probabilities, to check to what extent the optimal solution (set of facilities) depends on the failure probabilities. Table 6 displays the number of instances for which the set of optimal facilities obtained with two different intervals of failure probabilities was the same. From this table we can conclude that the set of optimal facilities depends quite a lot on the values of the failure probabilities, and more on their magnitude than on their variance, specially in the case of larger instances. The set of optimal facilities was the same for the four intervals of failure probabilities only in seven of the 30 original instances. In the remaining 23 instances, either two or three different sets of optimal solutions were obtained, depending on the considered intervals of probabilities. This reinforces our belief that even if failure probabilities are usually small, ignoring them and simply solving a pM instead of the $RpMF$ can be a big mistake.

6. Conclusions

In this paper we introduce different formulations to model a facility location problem where facilities may fail and service must be provided once the customer reaches the serving facility (*at-facility service*). Therefore, customers do not have information concerning facility failures before reaching them. In addition, we assume that, when failures occur, the sequence in which customers visit the facilities for being served is the one minimizing their expected total travel cost. For this model, we first present a compact formulation that uses variables with three indices and mixes binary and continuous variables. Then, for the particular case where all the failure probabilities are the same, we present a second formulation, rather intuitive, that only uses binary variables, at the price of using four index variables. These are used to trace the prospective paths of each of the customers. We compare both formulations and prove that in the case of homogeneous failure probabilities they are equivalent. Moreover, the computational experiments show that the first one provides tighter LP lower bounds. Both formulations require large computing times to solve even medium size instances. For this reason, we introduce a third formulation that heuristically approximates the exact solution with a much lower computing time and with high accuracy. Computational experiments show that this formulation gives extremely good solutions at a non-expensive computing time.

Further research on this topic includes the development of *ad-hoc* branch-and-cut algorithms for solving the exact formulations of the $RpMF$ model that allow to address larger size instances.

Acknowledgments

The research activities of the last two authors have been supported by projects FQM-5849 (Junta de Andalucía\FEDER) and MTM2010-

19576-C02-01, MTM2013-46962-C2-1-P (MICINN, Spain). The work of the first author has been supported by the Spanish Ministry of Economy and Competitiveness and ERDF funds through Project MTM2012-36163-C06-05. This support is gratefully acknowledged. Thanks are also due to the two anonymous referees and the editor for their valuable comments.

References

- Aboolian, R., Cui, T., & Shen, Z. J. M. (2013). An efficient approach for solving reliable facility location models. *INFORMS Journal on Computing*, 25(4), 720–729.
- Aksen, D., & Aras, N. (2012). A bilevel fixed charge location model for facilities under imminent attack. *Computers & Operations Research*, 39(7), 1364–1381.
- Albareda, M., Hinojosa, Y., Marin, A., & Puerto, J. (2015). When centers can fail: A close second opportunity. *Computers & Operations Research*. doi:10.1016/j.cor.2015.01.002.
- Alcaraz, J., Landete, M., & Monge, J. (2012). Design and analysis of hybrid metaheuristics for the reliability p -median problem. *European Journal of Operational Research*, 222(1), 54–64.
- Aydin, N., & Murat, A. (2013). A swarm intelligence based sample average approximation algorithm for the capacitated reliable facility location problem. *International Journal of Production Economics*, 145(1), 173–183.
- Benyoucef, L., Xie, X., & Tanonkou, G. (2013). Supply chain network design with unreliable suppliers: A lagrangian relaxation-based approach. *International Journal of Production Research*, 51, 6435–6454.
- Berman, O., Huang, R., Kim, S., & Menezes, M. B. C. (2007a). Locating capacitated facilities to maximize captured demand. *IIE Transactions*, 39(11), 1015–1029.
- Berman, O., Ivanovsky, E., & Krass, D. (2011). Optimal search path for service in the presence of disruptions. *Computers & Operations Research*, 38, 1562–1571.
- Berman, O., Krass, D., & Menezes, M. B. C. (2007b). Facility reliability issues in network p -median problems: Strategic centralization and co-location effects. *Operations Research*, 55(2), 332–350.
- Berman, O., Krass, D., & Menezes, M. B. C. (2009). Locating facilities in the presence of disruptions and incomplete information. *Decision Sciences*, 40(4), 845–868.
- Berman, O., Krass, D., & Menezes, M. B. C. (2013). Location and reliability problems on a line: Impact of objectives and correlated failures on optimal location patterns. *Omega*, 41, 766–779.
- Chen, G., Daskin, M. S., Shen, Z. J. M., & Uryasev, S. (2006). The α -reliable mean-excess regret model for stochastic facility location modeling. *Naval Research Logistics*, 53(7), 617–626.
- Church, R. L., Scaparra, M. P., & 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.
- Cui, T., Ouyang, Y., & Shen, Z. J. M. (2010). Reliable facility location design under the risk of disruptions. *Operations Research*, 58, 998–1011.
- Drezner, Z. (1987). Heuristic solution methods for two location problems with unreliable facilities. *Journal of the Operations Research Society*, 38, 509–514.
- Drezner, Z., & Hamacher, H. (2004). *Facility location: Applications and theory*. Springer.
- Gade, D., & Pohl, E. (2009). Sample average approximation applied to the capacitated-facilities location problem with unreliable facilities. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 259–269.
- Galindo, G., & Batta, R. (2013). Review of recent developments in OR/MS research in disaster operations management. *European Journal of Operational Research*, 203, 201–211.
- Guha, S., Meyerson, A., & Munagala, K. (2003). A constant factor approximation algorithm for the fault-tolerant facility location problem. *Journal of Algorithms*, 48, 429–440.
- Kulturel-Konak, S., & Konak, A. (2010). Simulation optimization embedded particle swarm optimization for reliable server assignment. In *Proceedings - Winter Simulation Conference* (pp. 2897–2906).
- Li, Q., Zeng, B., & Savachkin, A. (2013a). Reliable facility location design under disruptions. *Computers & Operations Research*, 40(4), 901–909.
- Li, X., Ouyang, Y., & Peng, F. (2013b). A supporting station model for reliable infrastructure location design under interdependent disruptions. *Transportation Research Part E*, 60, 80–93.
- Liberatore, F., Scaparra, M. P., & 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.
- Liberatore, F., Scaparra, M. P., & Daskin, M. S. (2012). Hedging against disruptions with ripple effects in location analysis. *Omega*, 40(1), 21–30.
- Lim, M., Daskin, M. S., Bassamboo, A., & Chopra, S. (2010). A facility reliability problem: Formulation, properties, and algorithm. *Naval Research Logistics*, 57, 58–70.
- Liu, Z., Guo, S., Snyder, L., Lim, A., & Peng, P. (2010). A p -robust capacitated network design model with facility disruptions. *Lecture Notes in Business Information Processing*, 46, 269–280.
- Morton, D. P., Pan, F., & Saeger, K. J. (2007). Models for nuclear smuggling interdiction. *IIE Transactions*, 39(1), 3–14.
- O’Hanley, J., & Church, R. (2011). Designing robust coverage networks to hedge against worst-case facility losses. *European Journal of Operational Research*, 209(1), 23–36.

- O'Hanley, J. R., Scaparra, M. P., & García, S. (2013). Probability chains: A general linearization technique for modeling reliability in facility location and related problems. *European Journal of Operational Research*, 230(1), 63–75.
- Pan, F., & Morton, D. P. (2008). Minimizing a stochastic maximum reliability path. *Networks*, 52, 111–119.
- Peng, P., Snyder, L., Lim, A., & Liu, S. (2011). Reliable logistics networks design with facility disruptions. *Transportation Research Part B: Methodological*, 45(8), 1190–1211.
- Qi, L., & Shen, Z. J. M. (2007). A supply chain design model with unreliable supply. *Naval Research Logistics*, 54, 829–844.
- Shen, Z. J. M., Shan, R., & Shang, J. (2011). The reliable facility location problem: Formulations, heuristics and approximation algorithms. *INFORMS Journal on Computing*, 23(3), 470–482.
- Snyder, L., & Daskin, M. S. (2005). Reliability models for facility location: The expected failure cost case. *Transportation Science*, 39(3), 400–416.
- Witlin, D. (February 17, 2006). Ambulance diverted each minute. *Boston Herald*.
- Zhang, Y., Berman, O., & Verter, V. (2009). Incorporating congestion in preventive healthcare facility network design. *European Journal of Operational Research*, 198(3), 922–935.