Reliability Engineering and System Safety 123 (2014) 73–80

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Robust facility location: Hedging against failures

Ivan Hernandez [a], Jose Emmanuel Ramirez-Marquez [a], Chase Rainwater [b], Edward Pohl [b], Hugh Medal [c]

[a] Stevens Institute of Technology, United States
[b] University of Arkansas, United States
[c] Mississippi State University, United States

ARTICLE INFO

Article history:
Received 8 March 2013
Received in revised form
22 October 2013
Accepted 31 October 2013
Available online 11 November 2013

Keywords:
Resilience
System safety
Facility location
Robustness
Multi-objective
Optimization

ABSTRACT

While few companies would be willing to sacrifice day-to-day operations to hedge against disruptions, designing for robustness can yield solutions that perform well before and after failures have occurred. Through a multi-objective optimization approach this paper provides decision makers the option to trade-off total weighted distance before and after disruptions in the Facility Location Problem. Additionally, this approach allows decision makers to understand the impact on the opening of facilities on total distance and on system robustness (considering the system as the set of located facilities). This approach differs from previous studies in that hedging against failures is done without having to elicit facility failure probabilities concurrently without requiring the allocation of additional hardening/protections resources. The approach is applied to two datasets from the literature.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

The Uncapacitated Facility Location Problem (UFLP) considers a set of demand points that need to be serviced from a set of potential facilities. The objective is to select the best subset of facilities to open in order to minimize total distance from demand centers to facilities [1]. Applications of the UFLP include the placement of warehouses [2] and disaster recovery centers [3], integrated circuit design [2] and clustering [4].

Because of today's globalized threats, such as deliberate sabotage and terrorist attacks [5], there has been renewed interest in analyzing the UFLP with a focus on vulnerability reduction. However, terrorist attacks and sabotage are not the only threat to facilities. Facilities are also subject to labor disruptions, change of ownership [6] and failure due to weather. Consider for example that in December 2012 the White House requested 60.4 billion US$ for response, recovery and mitigation related to Hurricane Sandy damage in all affected states. These resources included efforts to repair damages to homes and public infrastructure and to help affected communities prepare for future storms [7].

To address these previous issues, models have been presented in which resources are allocated to protect the most critical facilities, thus minimizing effects of worst-case disruptions [8]. Another study analyzed the Robust Maximum Covering Problem, where decision makers are able to design robust coverage networks. The study formulated the problem as a bilevel mixed integer program, where the objectives to maximize are the initial coverage and the coverage after the most critical facilities have failed [9]. Also, a multi-objective optimization model has been presented in which facilities are located with the objective to minimize day-to-day construction and operation costs and also with the objective of minimizing the expected total distance after facilities have failed [6]. The aforementioned model allows decision makers to obtain robust solutions by hedging against failures.

Although few companies would be willing to choose solutions with location and day-to-day transportation cost (total weighted distance) much greater than optimal just to hedge against occasional disruptions in their supply network, substantial improvements can often be obtained without large increases in day-to-day operating costs. By taking reliability into account at design time, one can find near-optimal solutions to the UFLP that are much more reliable [6]. However, accounting for facility reliability is a non-trivial process—impossible without accurate data.

When considering hedging against failures, one of the most important and timely problems in facility location is the issue of system resilience [10]. Optimizing system resilience allows distribution networks to regain operational performance after a disruption as quickly and efficiently as possible. The techniques presented in this work aim to advance our knowledge towards solving this

E-mail addresses: ihernand@stevens.edu (I. Hernandez), jmarquez@stevens.edu (J. Emmanuel Ramirez-Marquez), cer@uark.edu (C. Rainwater), epohl@uark.edu (E. Pohl), hugh.medal@msstate.edu (H. Medal).

0951-8320/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.ress.2013.10.006

important issue as it relates to the UFLP. Because the literature is awash with definitions of resilience in areas outside system robustness, there have been efforts to develop a unified definition [11]. In this paper, the definition of system resilience used is the time dependent ratio of recovery over loss. To evaluate the resiliency of a system, three measures need to be computed: (1) value of resilience (Я_{ϕ}), (2) cost of the resilience strategy (C_r) and (3) time to implement the resilience strategy (T_r). The formula for calculating the value of resilience at time t_r based on the objective function ϕ is [11]

$$\text{Я}_{\phi}(t_r) = \frac{\phi(t_r) - \phi(t_d)}{\phi(t_0) - \phi(t_d)} \tag{1}$$

Eq. (1) is obtained from the representation of system resilience as illustrated in Fig. 1. There are three aspects related to system resilience: reliability, vulnerability and recoverability. Accordingly, the system is in its original state at time t_0, suffers a disruption at time t_e and reaches its worse performance at time t_d. A resilience action is taken at time t_s and the system reaches its recovery status at time t_f (i.e. "bounces back"). The system performance is evaluated at time $t_r \in (t_d, t_f)$. A system can be in one of five states: (1) initial state (the system is performing as it is supposed to, due to its reliability), (2) the system can have a partial or full failure (due to its vulnerability), (3) the system can be in a disrupted state (while it is waiting to be repaired), (4) the system can be on recovery mode, trying to regain its original status (due to its resilience) and (5) the system reaches its final state (recovered). The resilience of the system is what allows a system to transition from a disrupted state to an operational state. Resilience might have a cost and time associated.

We claim that most of the papers found in the literature focus on the reliability aspect. That is, allocating resources to prevent the system from failing. Our focus is on the vulnerability aspect: under the assumption of failures, the proposed method is intended to minimize vulnerability as a means to hasten the recoverability of the system; our rationale is that by increasing the robustness of the system, resources can be better allocated to system recovery. It is important to mention that "system" in this paper is the set of open facilities. Also, throughout this paper, by failure of a facility we mean a facility that was selected for opening and it failed afterwards. Once a facility fails, it can no longer satisfy the demand of the demand centers assigned to it. Therefore, demand centers need to be re-assigned to their nearest open and functioning facility.

To illustrate the concept of resilience, consider Fig. 2, which illustrates how the efforts of the electric company PSE&G gradually restored power to customers in New Jersey [12]. The graph illustrates the effect of Sandy on the first day (October 29th 2012, when the disruptive event hit New Jersey) and also the recoverability effort behavior during ten days of recovery efforts. If the same electric network were to improve its resilience and the same storm were to hit again, the number of outages would be smaller (vulnerability reduction) and/or the restoration time shorter (recoverability improvement). Note that the diagram in (Fig. 1(b)) mimics the resilience process of PSE&G when the service function is the number of customers without power.

To adapt these concepts to the UFLP, we present a method that allows decision makers to trade-off the number of facilities to open, the total distance before failure and the total distance after failure. With our approach, decision makers can understand day-to-day performance trade-offs to obtain a good performance after occasional disruptions. Additionally, decision makers can understand the impact that the opening of additional facilities has on total distance. Our model is formulated as a bi-level binary program. In the first level we obtain the total distance without failure. In the second level, we obtain the worst-case total distance after failures. With these two total distances, the third and final step seeks to obtain solutions that allow decision makers to hedge against failures. It is important to note that this approach differs from previous studies in that hedging against failures is done without having to elicit facility failure probabilities concurrently without requiring the allocation of additional hardening/protections resources. Our list of contributions is the following:

1. A systematic approach to allocate facilities in such a way that in the event of failures, the total distance remains as minimum.
2. A decision making approach that allows to trade-off total distance without failures, total distance considering failures and number of facilities.
3. An approach towards increasing system resilience that does not depend on probabilities and does not require any extra protection costs.

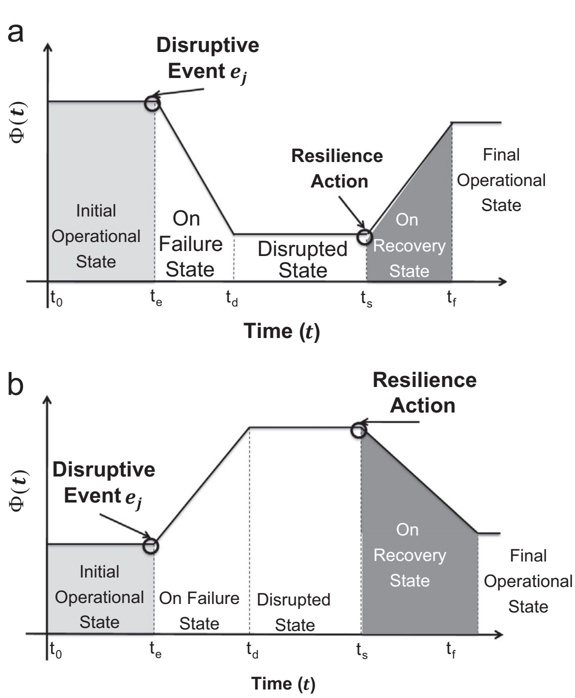

Fig. 1. System resilience diagram. (a) Decreasing values of service function $\phi()$ (e.g. the number clients processed). (b) Increasing values of service function $\phi()$ (e.g. the number of defective components).

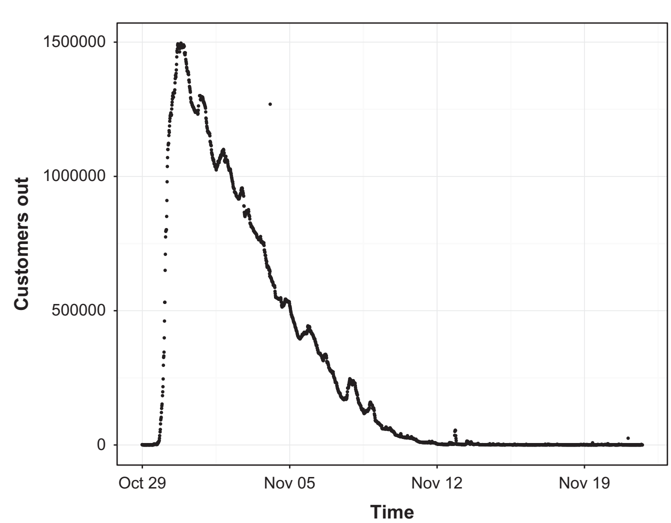

Fig. 2. Resilience of the electric company PSE&G in New Jersey during Sandy storm [12].

The rest of the paper is organized as follows. Section 2 describes the UFLP and the multi-objective bi-level problem for selecting robust facilities. Section 3 describes our three step solution approach that uses meta-heuristics (NSGA-II [13] and MOPSDA [14]). Section 4 presents our results using two standard datasets from the literature, the Swain [15] dataset and the London [16] dataset. Section 5 analyses the solutions and explains the options available for the Decision Maker through the proposed approach. Finally, Section 6 presents a summary of the main contributions of our paper.

2. Problem

2.1. List of key symbols

2.1.1. Sets

D	Set of demand centers
F	Set of all candidate facilities
G	Set of open facilities $\{i \in F \mid y_i = 1\}$
W	Set of open facilities that did not fail $\{i \in G \mid v_i = 0\}$
$\mathcal{P}(W)$	Power set of set W, i.e. all possible subsets of W

2.1.2. Parameters

f_i	Cost of opening a facility at location i $\{f_i = 1$ for all $i \in F\}$
c_{ij}	Cost of assigning demand center j to facility i

2.1.3. Decision variables

y_i	Binary decision variable that determines if facility i is open or not
x_{ij}	Binary decision variable that indicates if location j is assigned to facility i or not
u_{ij}	Binary decision variable that indicates if location j is assigned to facility i after failures have occurred
v_i	Binary decision variable that indicates if open facility i fails or not

2.2. Background

In the UFLP there is a set of locations for building a facility, where the cost of building at location i is f_i. There is also a set of demand centers that require service by a facility and if a demand center j is assigned to a facility at location i, a cost of c_{ij} that is proportional to the distance from i to j is incurred. The objective is to determine a set of locations at which to open facilities so as to minimize the total facility and assignment cost. Each facility can serve an unlimited amount of demand centers. Each demand center has a positive demand d_j that needs to be shipped to its assigned location. This problem is NP-hard and its mathematical formulation is as follows [1]:

$$\underset{\mathbf{y},\mathbf{x}}{\text{minimize}} \quad \sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij} \tag{2a}$$

$$\text{subject to} \quad \sum_{i \in F} x_{ij} = 1, \quad \text{for each } j \in D, \tag{2b}$$

$$x_{ij} \le y_i, \quad \text{for each } j \in D, \tag{2c}$$

$$x_{ij} \in \{0,1\}, \quad \text{for each } i \in F,\ j \in D, \tag{2d}$$

$$y_i \in \{0,1\}, \quad \text{for each } i \in F, \tag{2e}$$

where the objective function represents the addition of the set up cost ($f_i y_i$) and the assignment cost ($d_j c_{ij} x_{ij}$). The first constraint ensures that each demand center is assigned to a facility (e.g. distribution center). The second constraint ensures that demand centers are assigned only to open facilities. The last two constraints describe the binary nature of the decision variables.

2.2.1. Datasets

The main inputs for the FLP are the set of demand points and the set of candidate facilities. Fig. 3 shows the two main datasets use in this paper. Each graph shows the coordinates of the demand centers in the Cartesian plane (the size of each point is proportional to its demand). Additionally, each point is a potential location for opening a facility. The total distance between a demand center and an open facility is computed using the weighted Euclidean distance. Both datasets are used as benchmark in the location analysis literature [17,8]. Fig. 3(a) Swain dataset, where the data is drawn from a common base composed of 55 nodes identified by their (x, y) coordinates and their user population. The data approximates the distribution of air travelers by origin or destination for Washington, DC in 1960. Fig. 3(b) London dataset, which represents the location of gasoline stations and fire stations and where distances are based upon a road network.

2.3. Formulation of the robust facility location problem

The robust facility location problem is a facility location problem in which the objective is to seek solutions that perform well when parts of the system fail. In other words, the objective is to hedge against uncertainty in the solution itself. Unlike stochastic facility location models, which seek robustness to changes in demand or cost, robust models seek robustness to changes in the supply network itself [6]. As discussed in Section 1, current approaches for the robust facility location problem have focused on optimizing protection resources against interdictors or assume that facilities have a specified probability of failure [18,6]. However, we claim that because some of the failures have a High Impact and a Low Probability (HILP), in many instances there is not enough data to estimate the failure probability of the facilities, rendering the models with serious estimation gaps. In this section we present our formulation of the robust facility location problem. The novelty of our formulation is that: (1) hedging against failures is done without having to elicit facility failure probabilities, (2) availability of extra resources for protection is not necessary and (3) a multi-objective formulation is considered to identify trade-offs among solutions. Initially, the decision variable $\mathbf{x}$ determines the assignment of demand centers to facilities. However, we assume that some failures will occur and some demand centers will need to be re-assigned (because their assigned facility failed). The decision variable $\mathbf{u}$ indicates the post-failure assignment of demand centers to facilities that did not fail. Because we are initially interested in simultaneously having solutions that perform well without any failures and that perform well for all possible failures, our initial mathematical formulation is as follows:

$$\underset{\mathbf{y}}{\text{minimize}} \quad \sum_{i \in F} y_i f_i \tag{3a}$$

$$\underset{\mathbf{x}}{\text{minimize}} \quad \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij}, \tag{3b}$$

$$\underset{\mathbf{u}}{\text{minimize}} \quad \sum_{k \in Z} \sum_{j \in D} d_j c_{kj} u_{kj}, \quad \text{for each } Z \in \mathcal{P}(G) \tag{3c}$$

$$\text{subject to} \quad u_{kj} \le x_{kj}, \quad \text{for each } k \in Z, j \in D, \text{ for each } Z \in \mathcal{P}(G) \tag{3d}$$

$$\sum_{k \in Z} u_{kj} = 1, \quad \text{for each } j \in D, \text{ for each } Z \in \mathcal{P}(G) \tag{3e}$$

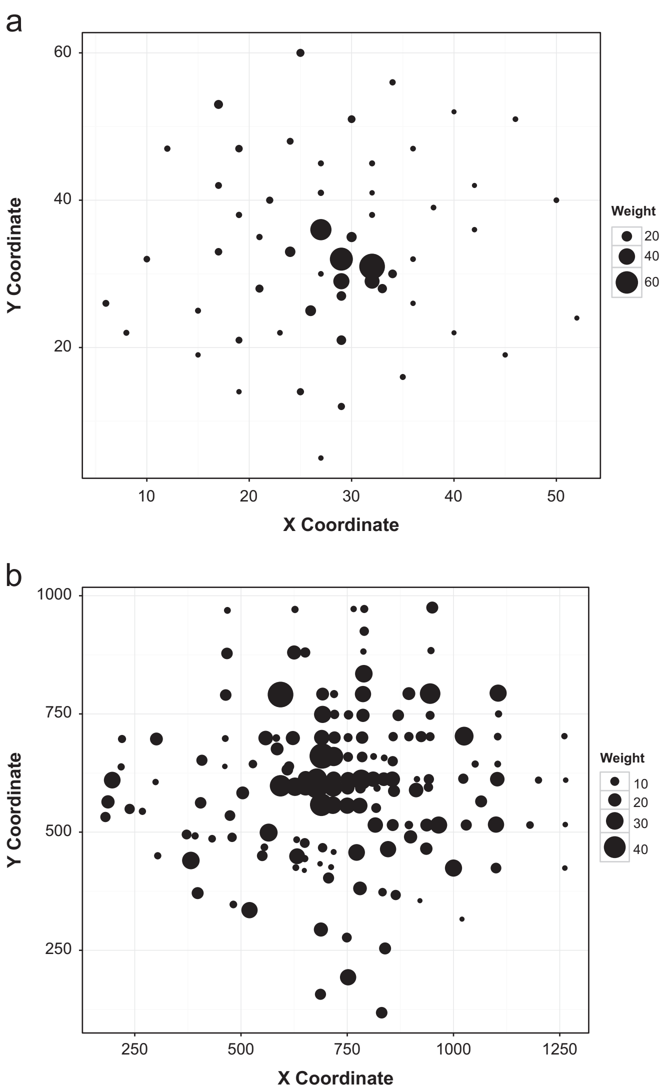

Fig. 3. Location datasets used in the experiments section. Each point represents a demand center and a candidate location for placing a facility. The size of each point is proportional to its demand. (a) Swain (55 Demand points) and (b) London, Ontario (150 Demand points).

$$u_{kj} \in \{0,1\}, \quad \text{for each } k \in Z, j \in D, \text{ for each } Z \in \mathcal{P}(G) \tag{3f}$$

In this formulation the constraints of formulation (2) still apply. The first objective function minimizes the cost of opening facilities. The second objective function minimizes the total distance without failures. The third objective function minimizes the total distance after failures, for all possible failure cases (e.g. 1, 2, 3 failures). Constraint (3d) ensures that demand centers are re-assigned to open facilities. Constraint (3e) ensures that demand centers are re-assigned. Constraint (3f) indicates the binary nature of the decision variables.

This problem formulation allows decision makers to simultaneously trade-off distance before failure, distance after failure and number of facilities to open. However, this formulation has two limitations. The first limitation is that the computation of all possible subsets of facilities ($\mathcal{P}(G)$) is intensive (even for small data sets of facilities). Therefore, the model might be impractical when a large set of facilities is considered and multiple failure scenarios are analyzed. The other limitation is this formulation considers a risk seeking decision maker: by considering all possible failure cases, there will be instances where solutions will be based on the failures of the least critical facilities (in terms of distance). If the decision maker is risk averse, a reformulation is needed. To clarify, from a risk seeking perspective, failure of the least critical facilities will yield minimal solutions with respect to distance after failures (the distance will increase as little as possible) while risk averse perspective considers worst-case scenarios (maximal distance after failures will always be dominated by the minimal distance after failures).

2.4. *Reformulation of the robust facility location problem*

In this section, we re-formulate the problem from the previous section and decompose it into 3 subproblems that need to be sequentially solved. The solution to the first subproblem represents the day-to-day transportation cost (total distance). The solution of the second subproblem represents worst-case transportation cost (total distance) after disruptions (solved for each solution found on the first subproblem). Finally, the solution of the third subproblem provides trade-offs between distance before failure and distance after failure (solved for each solution found on the second subproblem). The rationale for analyzing worst-case is that we are not making assumptions about the probabilities of failures nor the resources for protecting or attacking facilities. Therefore, the distance after failure provided by our approach represents an upper bound. The other advantage of decomposing the original problem into three subproblems is that the three subproblems need not be solved for all possible cases of facilities to open (e.g. 1, 2, 3 facilities to open). The subdivision allows the decision maker to explore the cases that are of interest to him/her and discard other cases if he/she wishes to do so. For example, instead of solving all three subproblems for all cases, the decision maker can select a desired number of facilities to open after analyzing the Pareto set (Pareto front) of the first problem. Afterwards, the focus of the two last problems can be based on a specified number of facilities. This flexibility is similar to the reference-point based multi-objective optimization [19,20].

2.4.1. *First subproblem*

The first subproblem is a bi-objective binary optimization problem

$$\underset{\mathbf{y}}{\text{minimize}} \quad \sum_{i \in F} f_i y_i \tag{4a}$$

$$\underset{\mathbf{x}}{\text{minimize}} \quad \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij}, \tag{4b}$$

where the first objective function minimizes the cost of opening facilities and the second objective function minimizes the total distance. This problem is the multi-objective version of the problem (2a). Transforming the classical facility location objective function into two objectives, allows decision makers to understand the impact that one objective has over the other (i.e. how opening more facilities reduces the total distance). As such, decision makers can select the appropriate trade-off. Constraints (2b)–(2e) still apply.

2.4.2. *Second subproblem*

The second subproblem finds the worst-case total distance after disruptions for each solution found on the first subproblem. The second sub-problem is multi-objective and binary, its formulation is

$$\underset{\mathbf{v}}{\text{minimize}} \quad \sum_{m \in G} f_m v_m \tag{5a}$$

$$\underset{\mathbf{u}}{\text{maximize}} \quad \sum_{k \in W} \sum_{j \in D} d_j c_{kj} u_{kj} \tag{5b}$$

$$\text{subject to} \quad v_m \geq y_m, \quad \text{for each } m \in G \tag{5c}$$

$$v_m \in \{0,1\}, \quad \text{for each } m \in G, \tag{5d}$$

$$u_{kj} \leq x_{kj}, \quad \text{for each } k \in W, j \in D \tag{5e}$$

$$\sum_{k \in W} u_{kj} = 1, \quad \text{for each } j \in D, \tag{5f}$$

where $\mathbf{v}$ is the decision variable concerned with the selection of the facilities that fail. Objective function (5a) minimizes the number of facilities that fail. Objective function (5b) maximizes the distance after failures. Constraints (5d) ensures that $\mathbf{v}$ is a binary decision variable. Constraints (5c) ensures that only open facilities can fail. Constraints (3d), (3e) and (3f) apply. This second subproblem formulation seeks to find the worst-case distance by minimizing the facilities that fail and maximizing the distance after failures. In other words, the formulation tries to find the most effective way (by failing facilities) to effect the highest increase in total distance.

2.4.3. *Third subproblem*

The third problem that needs to be solved provides the decision maker with the option of trading off day-to-day performance (lower bound) versus performance after disruption (upper bound). The mathematical formulation is

$$\underset{\mathbf{x}}{\text{minimize}} \quad \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij} \tag{6a}$$

$$\underset{\mathbf{u}}{\text{minimize}} \quad \sum_{k \in W} \sum_{j \in D} d_j c_{kj} u_{kj} \tag{6b}$$

$$\text{subject to} \quad \sum_{k \in W} \sum_{j \in D} d_j c_{kj} u_{kj} < \sum_{n \in T} \sum_{j \in D} d_j c_{nj} x_{nj}, \tag{6c}$$

where T represents an optimal set of facilities of the first sub-problem ($|T| = |G|$). Objective (6a) minimizes distance before failure, while objective (6b) minimizes distance after failure. Constraint (6c) ensures that the distance before failure of any additional trade-off solution is at least smaller than the distance after failure of a solution found on the first subproblem. The objective of this problem formulation is to obtain more trade-off solutions for each number of facilities to open. The solutions found can be worse than the existing solutions with respect to distance before failure, but not with respect to distance after failures. This subproblem is solved for each solution found on the second subproblem.

3. Solution approach

In this paper, we propose the use of Multi-Objective Evolutionary Algorithms (MOEAs) to solve the bi-level optimization problems (4) and (5). We choose MOEAs because they are able to deal with noncontinuous, non-convex and/or non-linear objectives/constraints, as well as problems whose objective function is not explicitly known (e.g. the output of Monte Carlo simulation) [14]. MOEAs are specially useful in combinatorial optimization problems, where optimal solutions might not be guaranteed and where the solution space is large [21]. These algorithms obtain near optimal solutions by efficiently exploring just a fraction of the entire solution space. MOEAs are based on the process of evolution where the best traits of a population are identified and used to generate the next generation or replace the population. Specifically we use two evolutionary algorithms NSGA-II [13], which comes implemented in various optimization libraries [22,23] and MOPSDA [14], due to its simplicity. We use two evolutionary algorithms in order to enhance our solution, given that the two algorithms might explore different parts of the solution space. For the third subproblem we use Ordinal Optimization [24].

Other methods for solving multi-objective optimization problems include weighted sum method, goal programming, lexicographic method and bounded objective function among others. However, each of these alternative methods have their own drawbacks [25]. For example, one of the most common approaches to Multi-objective optimization is the *Weighted Sum Method*, which combines all objective functions into a single utility function that needs to be optimized

$$\underset{\mathbf{x}}{\text{minimize}} \quad \sum_{i=0}^{k} w_i f_i(\mathbf{x}), \tag{7}$$

where w_i represents the relative importance of the objective function f_i for the decision maker and k represents the number of objective functions. The drawback of this approach is the selection of the weights. One possible solution is to assign higher weights to more important objectives, but if the importance of the objectives is not known, many different weights need to be tried. However, trying different weights does not guarantee that the Pareto set will be explored evenly [25]. Another popular approach is *Goal Programming*, in which each objective function has a target value to be achieved and the total deviation from the targets is minimized

$$\underset{\mathbf{x}}{\text{minimize}} \quad \sum_{i=0}^{k} |f_i(\mathbf{x}) - b_i|, \tag{8}$$

where b_i is the goal of the ith objective. However, the drawback of this method is that the b_i need to be selected and the method does not guarantee Pareto Optimal solutions [25].

Our approach has the following three steps (the entire source code and its documentation is available on the following URL: https://github.com/ivihernandez/facility-location):

1. Approximate the optimal Pareto set with NSGA-II and with MOPSDA for problem (4). Merge both Pareto sets into one. We use a chromosome in order to represent decision variable y, where position i has the bit y_i. The chromosome length is equal to the number of distribution centers ($|F|$). Element y_i is equal to 1 if the facility is open and zero otherwise. Each chromosome is evaluated with two figures of merit, the first one to obtain the number of facilities and the second one to obtain its total distance (Fig. 4).
2. For each of the solutions found in the previous step, we find the optimal Pareto set according to problem (5) with NSGA-II and with MOPSDA. Afterwards, we merge both Pareto sets into one. For this subproblem we use a binary chromosome to represent decision variable v, where position i has the bit v_i. The length of the chromosome is equal to the number of facilities that were open in the previous step. Element v_i is equal to one if the facility fails and zero otherwise. For each chromosome, two figures of merit are obtained. The first one is the number of facilities that fail, the second one is the total distance after the re-assignment (Fig. 5). Fig. 5 provides more information than Fig. 4 because it not only shows total weighted distance without failure (lower bound, in black), but also distance after worst-case failure (upper bound, in grey). The Decision Maker can know trade-off two between two objectives and also know how much his/her total distance will increase in case of failures.
3. At this stage of the solution approach, each solution has four objective function values associated with it: (1) number of facilities, (2) total distance, (3) number of failures and (4) total distance after failures. All these solutions belong to the final Pareto set S, where each solution is referred to as s_i. The third step uses Ordinal Optimization to solve problem (6) and works as follows. First, the number of facilities (p) and number of failures (r) for which the decision maker is interested are selected. The second step involves finding in the Pareto set S a solution s_j whose objective function values match those of the decision maker. In the third step we use Monte Carlo simulation to open p facilities randomly and to also fail r of them. We obtain the total distance after and before failures. Any solution found whose total distance after failure is smaller than the total distance after failure of s_j is

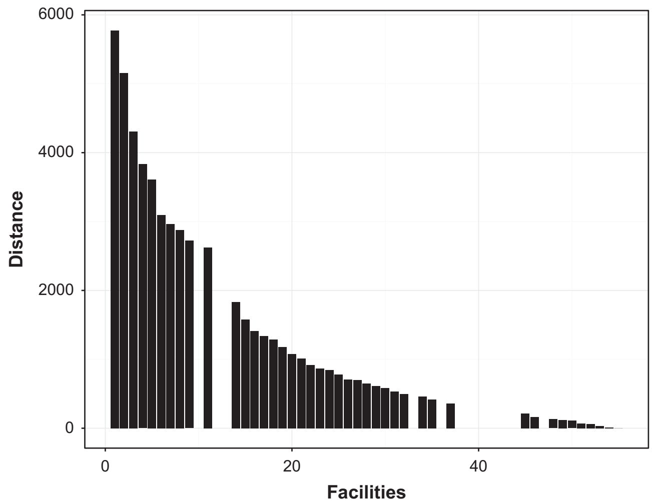

Fig. 4. Pareto set of problem (4) for to the Swain dataset. Decision makers can select the solution that best conforms to their needs by trading-off number of facilities to open (horizontal axis) versus total weighted distance to travel (vertical axis).

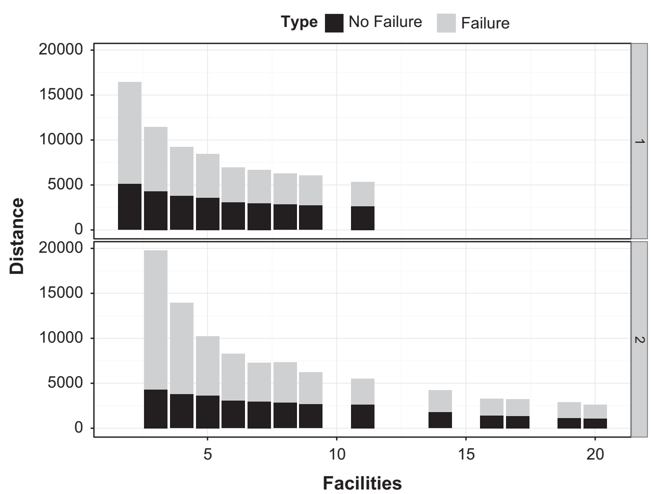

Fig. 5. Pareto set of problem (5) for the Swain dataset. Top: Considering a one facility worst-case failure. Bottom: Considering a two facility worst-case failure.

added to the final Pareto set S. For each number of facilities to open there are now multiple solutions (Fig. 6). Notice that in Fig. 6 the upper bound of problem (5) is used to filter certain candidate solutions. Solutions with a small increase in distance after failure, but with a "large" distance before failure are discarted. By "large" we mean that their distance without failure is larger than the corresponding upper bound found in the previous step.

4. Computational results

We performed two experiments, one with the Swain dataset and another one with the London dataset. All experiments were run in an Asus Laptop with an Intel Core i7 with 6 GB of RAM memory. In order to obtain the solutions we developed our program with Python and the library ECSPY [22]. Each experiment was run four times in order to obtain more accurate results. Tables 1 and 2 show the computational results obtained. For the Swain dataset the population size was 50 and the number of generations was also 50. For the London dataset the population size and the number of generations was set at 100. The column *time* indicates the average time of four runs (in minutes).

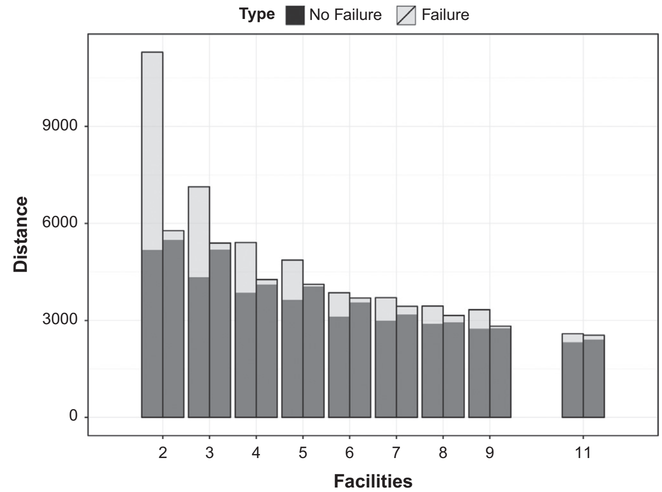

Fig. 6. Partial Pareto set of problem (6) for the Swain dataset. This figure provides more information than Fig. 5 because the Decision Maker can now trade-off between three objectives: number of facilities, distance before failure and distance after failure.

Table 1
Computational results for the first two subproblems of the Swain experiment.

Algorithm	Time (min)	Proportion (%)
NSGA-II	2.5	39.53
MO-PSDA	30.3	60.47

Table 2
Computational results for the first two subproblems of the London experiment.

Algorithm	Time (min)	Proportion (%)
NSGA-II	22.6	44.4
MO-PSDA	494	55.56

The column *proportion* indicates which percentage of the solutions obtained in the final Pareto set corresponds to each algorithm.

For the London dataset, Fig. 7 shows the pareto set of problem (4), Fig. 8 shows the Pareto set of problem (5) and Fig. 9 shows the Pareto set of Problem (6).

For the third step, the Monte Carlo simulation is configured to search for a specific number of facilities and a specific number of failures. The number of simulations was set to 10 000 and the computational time was 6.65 min for all the solutions shown of the Swain dataset and 35 min for all the solutions shown for the London dataset.

Based on the experiments performed, we concluded that there is a trade-off between using NSGA-II and MOPSDA. NSGA-II performs faster, but finds less solutions of the final Pareto set. The increase in computational time for MOPSDA comes from the fact that the Pareto set is larger and therefore in order to determine which elements are non-dominated takes longer. Another reason for using the heuristics was in order to speed up the computational results.

5. Discussion

The final solutions presented in Figs. 6 and 9 highlight the trade-offs available for the decision maker. For example, for the Swain dataset the decision maker can select between two different options. The first option has a low total distance before failure (5155.805) but a large total distance after worst-case failure

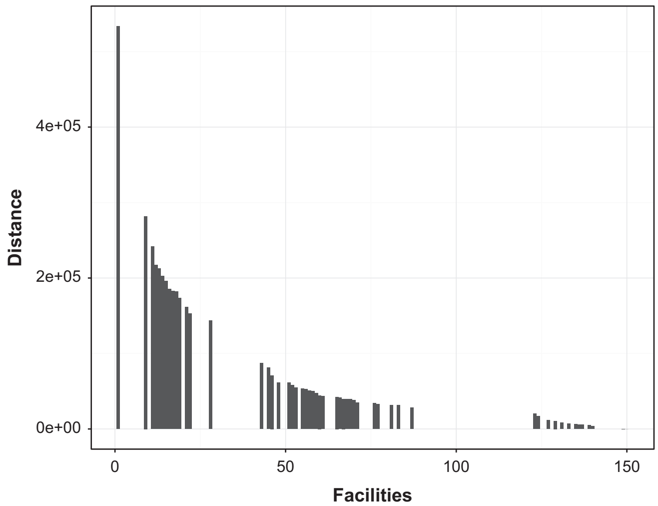

Fig. 7. Pareto set of problem (4) for the London dataset. Decision Makers can trade-off between two competing objectives: number of facilities to open and number of facilities to open.

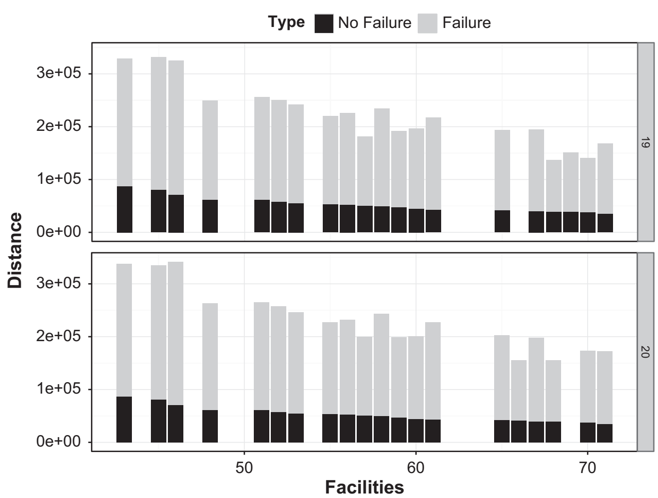

Fig. 8. Pareto set of problem (5) for the London dataset. Decision Makers can trade-off between two conflicting objectives and they can also know by how much their total distance will increase in case of failures. Top: 19 failures. Bottom: 20 failures.

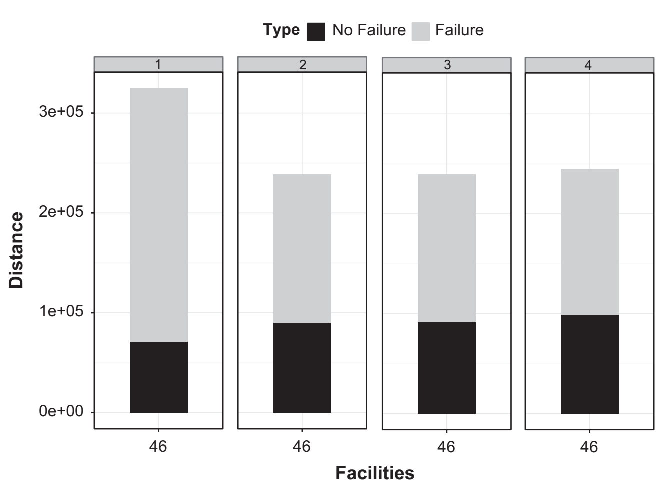

Fig. 9. Pareto set of problem (6) for the London dataset considering 46 facilities. For each number of facilities there are multiple solutions to choose, depending on the desired trade-off between distance before failure and distance after failure. Four solutions are obtained when the number of failures to consider is fixed at 19.

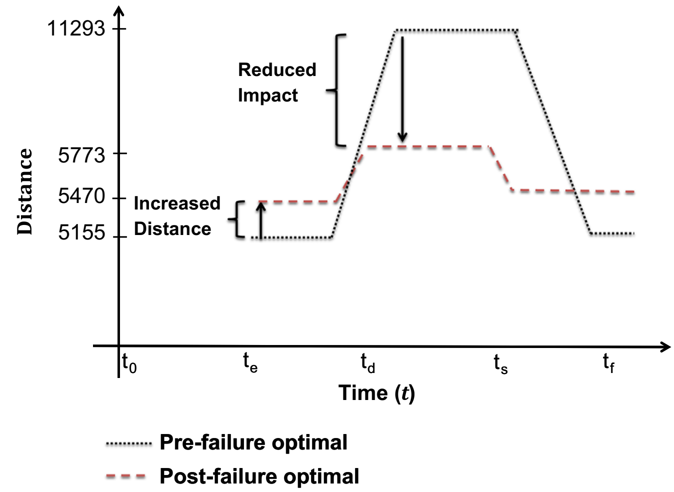

Fig. 10. Illustration of two trade-off solutions available for the Decision Makers. The solid line represents a solution in which distance before failure is the smallest (but the distance after failure is the largest). The dashed line represents a solution in which distance before failure is the largest (but the distance after failure is the smallest).

(11 293.271). The total distance increases by 119%. The second option sacrifices the original total distance with an increase of 6% (5470.473). However, after failures have occurred this second option has a total distance of 5773.96—an increase of roughly 12% from the original optimal solution but a 51% decrease from the original cost after worst-case failure. Surely, a significantly appealing option for a risk averse decision-maker. For the London dataset, four trade-off solutions are presented when deploying 19 facilities see Fig. 9). To better understand the different options available for the decision maker, the trade-off between two or more competing alternatives is presented from a resilience perspective. This can be appreciated in Fig. 10. Fig. 10 shows the distances taken from the Swain dataset when the number of facilities to open is one (the time component is added for illustration purposes).

5.1. Review of contributions and how they are met

1. *A systematic way to allocate facilities in a robust manner*. Through Sections 2.4 and 3 we have provided a framework that can be applied in order to allocate facilities in a robust way. Because the proposed solution approach is based on heuristics it works with any distance function and can scale with respect to the number of facilities.
2. *A decision making approach that allows a decision maker to simultaneously trade-off distance before failure, distance after failure and number of facilities*. Section 2.4 presents how the three objectives can be optimized.
3. *An approach towards increasing system resilience*. The approach presented is related to System Resilience, given that having a set of robust facility locations means that the distance after failure will be as small as possible, thus minimizing the impact and possibly facilitating the recovery in terms of cost or/and time. The relationship between resilience and our framework is explained in Section 1.

6. Conclusions and future work

We have presented an approach that allows decision makers to systematically deal with the location of facilities in a robust manner, thus allowing the transportation network to operate efficiently before and after failures. To our knowledge, this is the first approach to show from a multi-objective perspective how the transportation network can be more resilient without using any additional protection budget or redundancy and without assuming failure probabilities. Our problem formulation allows decision makers to simultaneously trade-off total distance before failure, total distance after failure and number of open facilities. For future work one priority would be to compare our solution approach based on heuristics with direct methods in terms of the quality of the solutions and the computational time needed to obtain them. Another priority would be to solve the capacitated facility location problem (where each facility can handle a limited amount of demand).

References

[1] David B. Shmoys, Éva Tardos, Karen Aardal. Approximation algorithms for facility location problems. In: Proceedings of the 29th annual ACM symposium on theory of computing; 1997. p. 265–74.
[2] Boyd S, Vandenberghe L. Convex optimization. Cambridge University Press; 2004.
[3] Dekle Jamie, Lavieri Mariel S, Martin Erica, Emir-Farinas Hülya, Francis Richard L. A Florida county locates disaster recovery centers. Interfaces 2005;35:133–9.
[4] Guha S, Khuller S. Gredy strikes back: improved facility location algorithms. In: Proceedings of the ninth annual ACM-SIAM symposium on discrete algorithms; 1998.
[5] Bricha N, Nourelfath M. Critical supply network protection against intentional attacks: a game-theoretical model. Reliab Eng Syst Saf 2013;119:1–10.
[6] Snyder LV, Daskin MS. Reliability models for facility location: the expected failure cost case. Transp Sci 2005;39:400–16.
[7] Zients JD. Transmittal letter from the deputy director for management of omb to the congress. Executive Office of the President. Office of Management and Budget; 2012.
[8] Scaparra MP, Church RL. A bilevel mixed-integer program for critical infrastructure protection planning. Comput Oper Res 2008;35:1905–23.
[9] O'Hanley JR, Church RL. Designing robust coverage networks to hedge against worst-case facility losses. Eur J Oper Res 2010;209:23–36.
[10] Losada C, Scaparra MP, O'Hanley JR. Optimizing system resilience: a facility protection model with recovery time. Eur J Oper Res 2011;217:519–30.
[11] Henry D, Ramirez-Marquez J. Generic metrics and quantitative approaches for system resilience as a function of time. Reliab Eng Syst Saf 2012;99:114–22.
[12] Cox Amanda, Keller Josh. Where the power is out and returning across the northeast. ⟨http://www.nytimes.com/interactive/2012/10/31/ny%region/where-the-power-is-out-and-returning.html?hp⟩, 2012.
[13] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Trans Evol Comput 2002;6:182–97.
[14] Rocco CM, Ramirez-Marquez JE, Salazar DE. Bi and tri-objective optimization in the deterministic network interdiction problem. Reliab Eng Syst Saf 2010;95:887–96.
[15] Swain RW. A decomposition algorithm for a class of facility location problems [Ph.D. thesis], Cornell University, Ithaca, NY, USA; 1971.
[16] Goodchild M, Norohona V. Location-allocation for small computers. Department of Geography, University of Iowa; 1983.
[17] Snyder LV, Shen Z-JM. Fundamentals of supply chain theory. Wiley; 2011.
[18] Losada C, Scaparra MP, Church RL, Daskin MS. The stochastic interdiction median problem with disruption intensity levels. Ann Oper Res 2012;201:345–65.
[19] Reference Point Approaches and Objective Ranking, Dagstuhl Seminar Proceedings.
[20] Deb K, Sundar J, Udaya Bhaskara Rao N, Chaudhuri S. Reference point based multi-objective optimization using evolutionary algorithms. In: International journal of computational intelligence research, Springer-Verlag; 2006. p. 635–42.
[21] Smith JC, Taşkın ZC. A tutorial guide to mixed-integer programming models and solution techniques. In: Lim GJ, Lee EK, editors. Optimization in medicine and biology. Taylor and Francis, Auerbach Publications; 2008.
[22] Garrett Aaron Lee. Evolutionary computations in python (ecspy). ⟨http://code.google.com/p/ecspy/⟩, 2011.
[23] Cahon S, Melab N, Talbi E-G. Paradiseo. ⟨http://paradiseo.gforge.inria.fr/⟩, 2011.
[24] Ho Y-C. An explanation of ordinal optimization: soft computing for hard problems. Inf Sci 1999;113:169–92.
[25] Marler R, Arora J. Survey of multi-objective optimization methods for engineering. Struct Multidiscip Optim 2004;26:365–9.