A two-level facility location and sizing problem for maximal coverage

Mumtaz Karatas[a,b,*], Abdullah Dasci[c]

[a] *Industrial Engineering Department, Naval Academy, National Defense University, Istanbul 34940, Turkey*
[b] *Industrial Engineering Department, Bahcesehir University, Istanbul 34353, Turkey*
[c] *Sabanci Business School, Sabanci University, Istanbul 34956, Turkey*

ARTICLE INFO

ABSTRACT

This paper presents a two-stage hierarchical location problem for systems where the lower level facilities act as the first points contact for the customers while the upper level facilities act as suppliers of the lower level facilities that either serve them or provide advanced services to customers. Furthermore, more recent and realistic coverage constructs such as gradual and cooperative covering are included in our setting. Although our problem can be applicable in various settings, the most fitting application is in wireless telecommunication networks to determine the location of base stations and mobile switching centers. We have developed two competing formulations for the problem, each of which involve nonlinear components that are difficult to deal with. We then develop their respective linearizations and tested their performances. These formulations are solved by commercial optimizers for a set of reasonably large problem instances and it is found that majority of the problems can be solved within a maximum of 10% optimality gap within a short time.

1. Introduction

In this paper we present a two-level facility location and sizing problem for maximizing expected demand coverage. At the lower level, there are branches that are the first contact with the customers and it is the proximity and the size of the branches that determine customers' attraction and hence, the expected demand coverage. At the upper level, there are centers, depending on a particular application, that either serve the branches and/or provide advanced services to the customers that the branches are not equipped to do. The problem is to find the location and the size of these facilities to maximize the expected demand coverage while satisfying a budget constraint.

This problem belongs to the large class of problems known as hierarchical location which has a long history and a sizable literature. According to Narula's (1984) classification, our model is a capacitated "coherent" hierarchical location problem with "successively exclusive" (i.e., non-nested) service with "referral" under covering objective. In this problem, the services provided at the lower level (branches) and the upper level (centers) are different, i.e., non-nested. Also, customers do not go to the upper level facilities without going thorough a lower level facility first, i.e., "referred". Finally, we assume a coherent structure; that is, customers who visit the same lower level facilities also visit the same upper level facility, e.g., single-sourcing in distribution context. This structure is applicable to some systems where

the centers only serve or supply the branches and/or do provide services to customers as well.

Although we utilize a terminology from retail setting, i.e., referring to facilities as "branches" and "centers", our problem can be applied in a wide variety of settings. One of the most fitting application areas of our model is in wireless telecommunication networks (Skorin-Kapov, Skorin-Kapov, & Boljunčic, 2006). In these networks, base stations are the first-contact points with the customers. The quality of transmission and hence, customers' service experience and eventual attraction depends on their proximity to as well as the power of the base-stations. Each base-station is connected to a mobile switching center via cable, which connects the cellular system to the wired public phone and internet systems. Hence, while the size (as a proxy for power) and the proximity of base stations (i.e., branches) to customers determine the demand capture, the size of the mobile switching centers (i.e., centers) makes sure that there is sufficient backbone capacity to satisfy the transmission demand.

There are a number of other distinguishing features and assumptions of our problem. First, we consider gradual coverage rather than binary coverage. That is, customers may be covered partially, which indicates that some proportion of customers at a demand node prefer to use the services while others do not. Second, we consider cooperative coverage where customers are not necessarily captured by the closest branch, but they may also be captured by some other branches that are

---
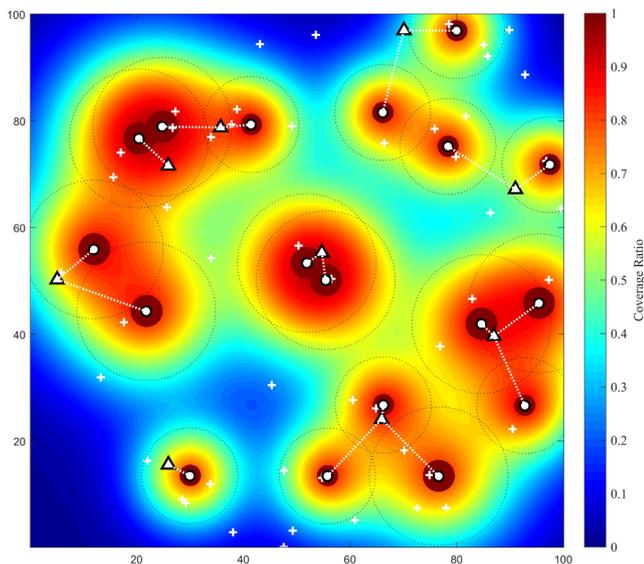
Fig. 1. An exemplary solution for the two-level facility location problem. In the figure, demand nodes, center locations, and branch locations are denoted by $+$, $\triangle$, $\circ$ signs, respectively. Dotted straight lines represent center-branch assignments.

further away but within the coverage distance. Both of these assumptions are well-justified in practice and have been around in the literature for some time (see for example, Berman, Drezner, & Krass, 2010 and Berman, Drezner, & Krass, 2018 2018 for further information). We also consider the size of facilities as decisions and assume that there are finite sizing options. Finally, we do *not* consider a "transportation cost" that depends on the volume of shipments between the centers and the branches, but rather we consider an "assignment cost" between them. This is a restriction that needs to be overcome in the future, but as we will see shortly, it is well-justified in a number of application areas. For example, in the wireless telecommunication networks, the transmission cost is essentially determined by the investment made for the cable connection between the base station and a mobile switching center and it is largely independent of the flow. An example solution of our problem is depicted in Fig. 1 where there are open branches that are assigned to some open centers. The coverage proportion of customer nodes are also represented by intensity of the colors, which is from full-coverage to no coverage.

There are many two-level hierarchical systems that satisfy the three key assumptions we make and naturally fit into our setting. The first assumption is the successively exclusive service, i.e., non-nested service. In many systems, upper level facilities provide different service than lower levels or they only serve the branches (which is also some form of providing *different* service). Second assumption is the coherent structure, which is also well-practiced in many public services where countries are divided into regions, regions into counties, counties into cities, etc. Coherent structure is also well-justified in many commercial systems as "single-sourcing" is practiced quite widely. Finally, there must either be a fixed assignment cost as in the wireless telecommunication networks or zero such cost. In health care and educational systems, if and when, a customer is referred to the higher level facility, the transportation cost is borne by the customer and hence, this assumption is satisfied. In some commercial cases where the volume of shipments is very small, such as banks and certain service systems, transportation cost can be considered largely fixed, i.e., does not depend on the volume, but only depends on the distance between the facilities and the frequency of service, which also satisfy this assumption. Hence, it is fair to conclude that, although the most fitting application area is in telecommunication networks, our model can also be applied to some public services such as non-nested health care systems

like blood collection/processing and medical testing systems, educational systems as well as to some service systems such as bank logistics systems and select retail systems.

Although our work belongs to a class of hierarchical location problems due to its three particular assumptions, it is also related to multi-level or multi-echelon facility location models. However, in great majority of these models the primary objective is usually some form of cost minimization with fixed charges and variable transportation cost (see for example, Ortiz-Astorquiza, Contreras, & Laporte (2018) for a recent comprehensive review). Our setting is rather different from this literature as our objective is to maximize demand coverage in settings where variable logistics costs are largely negligible, which brings our work closer to the hierarchical location literature.

Today there exists a vast literature on hierarchical location models, which can be traced back as early as to 1970s (see Narula, 1984 (1984) for an earlier review). Those with some type of coverage objective have been studied mostly in health care systems (see for example, Afshari & Peng, 2014 and Güneş & Nickel, 2015 for reviews of such models), but there are also growing number of papers in other application areas. The more recent examples include facility location in banking (Min & Melachrinoudis, 2001), design of geo-location systems for search and rescue operations (Chan, Mahan, Chrissis, Drake, & Wang, 2008), response center location to combat effects of some chemical, biological, or nuclear accidents or attacks (Paul, Lunday, & Nurre, 2017), earthquake shelter location (Li, Zhao, Huang, & Hu, 2017), distribution center location in disaster management (Li, Ramshani, & Huang, 2018), court relocation (Teixeira, Bigotte, Repolho, & Antunes, 2019), and in last-mile delivery system design (Chauhan, Unnikrishnan, & Figliozzi, 2019).

Most earlier works and those reviewed above assume binary covering and usually consider maximal coverage or set coverage type of objectives (please see the comprehensive review papers by Şahin & Süral (2007) and more recently, by Farahani, Hekmatfar, Fahimnia, & Kazemzadeh (2014)). They also usually consider coverage at all levels, whereas due to the particular application setting, we envision upper level facilities as the centers that serve the lower level facilities that actually interact with the customers directly. The main contribution of our work is in considering gradual and cooperative coverage in such a class of hierarchical location problems that existing models cannot adequately address.

Towards this end we have developed two nonlinear integer models. The models differ with respect to the decision variable sets they use. One utilizes three variable sets with double indices while the other utilizes two variable sets one with double and one with triple indices. In both models, however, the major difficulty comes from the nonlinearity of the objective function that involves the representation for gradual and cooperative coverage of customers, which is quite tedious. We linearize the models by adopting a special technique used by Morton, Pan, and Saeger (2007), Salmerón (2012), and Karatas (2018). Nonlinear and linear integer models are solved by DICOPT and Gurobi, respectively on large set of carefully designed test instances. Not surprisingly, owing to the complexity of the nonlinear terms, the nonlinear models could not be solved effectively. There is considerably more success in solving the MILPs and although the difference is not substantial, the formulation with the double indexed variables performed generally better. This was slightly surprising to us, as starting with Geoffrion and Graves (1974) usually longer indexed formulations have more success.

The rest of the paper is organized as follows: In the next section we give a detailed explanation of the assumptions and model preliminaries. Section 3 presents the four formulations; two nonlinear ones and their linearizations. In Section 4, we explain the generation of the instances for the numerical experiments and we report on the results of these experiments. Section 5 concludes the paper with a few remarks.

2. Assumptions and preliminaries

We consider a set of demand nodes $i \in I$, each endowed with a weight ω_i that reflects its relative importance or the maximum amount of demand generated over a period of time at that node. There are finite sets of candidate locations for the branches and the centers represented by $j \in J$ and $k \in K$, respectively. Adopting the discrete location framework, we assume that all candidate locations and demand nodes are fixed and known. In addition to the locations of facilities, the problem also considers the selection of the sizes of these facilities. We assume that each location can support a set of sizes for its facility. That is, there is a set of branch sizes $m \in M_j$ that can be opened at location $j \in J$, and similarly, there is a set of center sizes $n \in N_k$ that can be opened at location $k \in K$.

In our problem setting, there is a fixed budget B for installing and operating the facilities. Each type of facility has an installation cost that varies with respect to its size and location. We define b_{jm} as the fixed cost of installing a branch of size $m \in M_j$ at location $j \in J$ and c_{kn} as the fixed cost of installing a center of size $n \in N_k$ at location $k \in K$. We assume that there is "single-assignment", that is, a branch can be served from only one center. We define a fixed assignment cost a_{jk} to assign branch $j \in J$ to center $k \in K$. Note that this cost is independent of the sizes or potential flow between facilities, but only depends on the locations. As mentioned before, this assumption is well-justified in the design of cellular networks as well settings where the volume of the shipment is not a major factor. Also in cases where the customer incurs the transportation, this cost can be negligible or consists only of setting up the connection.

Here we adopt a partial or gradual coverage that depends on the distance between the demand nodes and the branch locations as well as the sizes of the branches. In the binary coverage (also known as the deterministic coverage) model, a demand node is assumed to be completely covered if the distance to its closest facility is less than a pre-specified value. This model is especially desirable for analytic derivation, but not always applicable to most real-world location-allocation problems (Berman et al., 2010 & Yang et al., Yang, He, Li, Chen, & Sun, 2015). For example, proximity to base stations and their sizes (as a proxy for power) are major determinants of customer satisfaction, which essentially determines their attraction and hence, the coverage. Similarly, in retail settings, proximity of the branches as well as their size as a proxy for variety are the main determinants for customers' attraction.

In this study, we adopt the gradual coverage concept from Berman, Krass, and Drezner (2003) where a demand can be fully covered, partially covered, or not covered. Towards this end, let d_{ij} denote the distance between demand node $i \in I$ and a branch at $j \in J$. We consider the demand of that node to be.

(i) fully covered by that branch if it lies within the minimum critical distance $\bar{d}_m$ of the branch with size $m \in M_j$, i.e. $d_{ij} \leqslant \bar{d}_m$,
(ii) not covered if the branch is not within the maximum critical distance $\bar{\bar{d}}_m$, i.e. $d_{ij} > \bar{\bar{d}}_m$, or
(iii) partially covered if it lies between the distances $\bar{d}_m$ and $\bar{\bar{d}}_m$, i.e. $\bar{d}_m < d_{ij} \leqslant \bar{\bar{d}}_m$,

which is represented by a non-increasing function of distance d_{ij} denoted as $f_m(d_{ij}) \in [0, 1]$, $\forall\, i \in I$, $j \in J$, $m \in M_j$. Hence, proportion of demand at node $i \in I$ covered by a branch at $j \in J$ with size $m \in M_j$ is given by

$$p_{ijm} = \begin{cases} 1, & d_{ij} \leqslant \bar{d}_m, \\ f_m(d_{ij}), & \bar{d}_m < d_{ij} \leqslant \bar{\bar{d}}_m, \; \forall\, i \in I, \; j \in J, \; m \in M_j. \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

It is also expected that a branch's coverage performance is positively related to its size, i.e. larger stores attract more customers. Therefore, we further assume that p_{ijm} is non-decreasing with increasing m for any given distance.

Different from the traditional modeling assumption that a demand can only be covered by a single (and generally the closest) facility, we adopt the joint or cooperative coverage concept which allows demand sharing, i.e. multiple branches contribute to the coverage of the same demand node. Joint coverage concept is also well-founded in practice (Berman et al., 2010 & Berman et al., 2018). Assuming independent coverage performance for each branch, for a particular demand node $i \in I$, we compute its overall coverage proportion P_i as

$$P_i = 1 - \prod_{j \in J} \prod_{m \in M_j} (1 - p_{ijm}). \tag{2}$$

The overall coverage is based on the assumption that a customer at a node can patronize a branch within its coverage radius with a certain probability (i.e., coverage proportion) and the choice of branches are independent of each other. Hence, for example if there are two facilities which attract, lets say 50% of demand at a node, their combined coverage would be 1-(1–0.5)(1–0.5) = 0.75.

As we have seen above, the major role the capacity of branches plays is on the total expected demand coverage. However, the centers must also be equipped with sufficient capacity to handle the demand at the branches. In the current development, we have information on the total demand covered but we have not made any specification about how the demand is split among the branches. Instead, we use the branch sizes as a proxy for the capacity requirement at the centers that the branches are assigned to. Hence, a branch at $j \in J$ with size $m \in M_j$ is assumed to have a capacity requirement of γ_m from center it is connected to and each open center must be assigned sufficient capacity to satisfy those requirements. Finally, we also assume that a branch can be serviced from predetermined subset $K_j \in K$ of alternative center locations, which are presumably closer than some threshold distance. We impose this requirement to ensure that branches can be served on a timely manner, but if all centers can service branch j, then we can simply set $K_j = K$.

3. Formulations

Based on the aforementioned assumptions we formulate two non-linear integer models for the problem, each followed by their respective linearizations. We first like to summarize the common notation before moving to the formulations.

Indices and Sets

$i \in I$:	Set of demand nodes,
$j \in J$:	Set of candidate branch locations,
$k \in K$:	Set of candidate center locations,
$K_j \subset K$:	Subset of centers that can serve branch $j \in J$,
$J_k \subset J$:	Subset of branches that can be served by center $k \in K$,
M_j:	Set of branch size levels that can be installed at location $j \in J$,
N_k:	Set of center size levels that can be installed at location $k \in K$.

Parameters

ω_i:	Weight of demand point $i \in I$,
B:	Available budget,
a_{jk}:	Fixed cost of assigning branch $j \in J$ to center $k \in K$,
b_{jm}:	Cost of installing a branch of size $m \in M_j$ at location $j \in J$,
c_{kn}:	Cost of installing a center of size level $n \in N_k$ at location $k \in K$,
γ_m:	Shipment requirement of a branch of size $m \in \bigcup_{j \in J} M_j$,
Q_n:	Capacity of a center of size level $n \in \bigcup_{k \in K} N_k$,
p_{ijm}:	Proportion of demand covered at node $i \in I$ by a branch of size $m \in M_j$ installed at location $j \in J$.

3.1. An Integer Non-Linear Programming (INLP) Formulation

For the first formulation we define three sets of decision variables; first two are for the location and sizing of the facilities and the third is for the assignment of branches to centers. These definitions will be followed by the first nonlinear formulation.

Decision Variables:

$$x_{jm} = \begin{cases} 1, & \text{if a branch of size } m \in M_j \text{ is opened at location } j \in J, \\ 0, & \text{otherwise,} \end{cases}$$

$$y_{kn} = \begin{cases} 1, & \text{if a center of size } n \in N_k \text{ is opened at location } k \in K, \\ 0, & \text{otherwise,} \end{cases}$$

$$z_{jk} = \begin{cases} 1, & \text{if the branch at location } j \in J \text{ is connected to the center at } k \in K_j, \\ 0, & \text{otherwise.} \end{cases}$$

$$\max z_{INLP1} = \sum_{i \in I} \omega_i \left[1 - \prod_{j \in J} \prod_{m \in M_j} (1 - p_{ijm} x_{jm}) \right] \tag{3a}$$

subject to

$$\sum_{m \in M_j} x_{jm} \leqslant 1, \ \forall j \in J \tag{3b}$$

$$\sum_{n \in N_k} y_{kn} \leqslant 1, \ \forall k \in K \tag{3c}$$

$$z_{jk} \leqslant \sum_{m \in M_j} x_{jm}, \ \forall j \in J, k \in K_j \tag{3d}$$

$$z_{jk} \leqslant \sum_{n \in N_k} y_{kn}, \ \forall j \in J, k \in K_j \tag{3e}$$

$$\sum_{m \in M_j} x_{jm} = \sum_{k \in K_j} z_{jk}, \ \forall j \in J \tag{3f}$$

$$\sum_{n \in N_k} y_{kn} \leqslant \sum_{j \in J_k} z_{jk}, \ \forall k \in K \tag{3g}$$

$$\sum_{m \in M_j} x_{jm} \leqslant \sum_{k \in K_j} \sum_{n \in N_k} y_{kn}, \ \forall j \in J \tag{3h}$$

$$\sum_{j \in J} \sum_{m \in M_j} b_{jm} x_{jm} + \sum_{k \in K} \sum_{n \in N_k} c_{kn} y_{kn} + \sum_{j \in J} \sum_{k \in K} a_{jk} z_{jk} \leqslant B \tag{3i}$$

$$\sum_{j \in J_k} \sum_{m \in M_j} \gamma_m z_{jk} x_{jm} \leqslant \sum_{n \in N_k} Q_n y_{kn}, \ \forall k \in K \tag{3j}$$

$$x_{jm} \in \{0, 1\}, \ \forall j \in J, m \in M_j \tag{3k}$$

$$y_{kn} \in \{0, 1\}, \ \forall k \in K, n \in N_k \tag{3l}$$

$$z_{jk} \in \{0, 1\}, \ \forall j \in J, k \in K_j \tag{3m}$$

The objective function given by (3a) aims to maximize total expected demand coverage. Constraint sets (3b) and (3c) ensure that an installed facility is designed with a single size. Constraint sets (3d) and (3e) guarantees that a branch can be assigned to a center only if both are open. Constraint set (3f) is a technical constraint which ensures that each open branch must be assigned to an open center and constraint set (3g) ensures that each center is connected to at least one branch. The latter constraint set essentially forces a center to be closed if there is no assignment to it. It is actually a redundant constraint because at the optimal solution no such center would be opened; nonetheless, it is a valid inequality that may help speed up the solution method. Constraint set (3h) ensures that if a branch is opened, then, a center than can serve it must also be opened. Constraint (3i) is the budget constraint, which restricts the installation and the assignment expenditures. Constraint set (3j) is a nonlinear constraint which ensures that the installed capacity of a center satisfies the total demand of its connected branches. Constraint sets (3k)–(3m) declare variable types.

3.2. Mixed integer linear programming (MILP) formulation

Owing to the difficulty of solving nonlinear integer models, we now reformulate the problem as a mixed integer linear model. In our original formulation, the objective function (3a) and the constraint set (3j) are nonlinear. Linearization of the constraint set is rather straightforward and will be given shortly. Linearization of the objective function requires some elaboration and will be presented subsequently.

We linearize constraint (3j) by defining a new continuous decision variable $h_{jkm} \in [0, 1]$, $\forall j \in J, k \in K_j, m \in M_j$ and switching constraint (3j) with constraints (4n) and (4o) in the MILP1 (see below). We further improve the formulation by adding two valid inequalities (4p) and (4q), which are not needed, but expected to speed up the solution. Note that even though we define the new variable as a continuous one, it will take the values of 0 or 1 in the optimal solution.

Next, we linearize the objective function (3a) by adopting a special technique used by Morton et al. (2007), Salmerón (2012), Karatas (2017, 2018). This technique creates a network-like structure by representing each candidate location as a node and the proportion of the *uncovered* demand as arcs. It uses the fact that the final outcome is independent of the order of the multiplication in Eq. (2). The idea of this technique can be simplified as follows: First, consider a particular customer. This customer "visits" each node (a location) and depending on whether a branch is opened or not and its size, a proportion of customer demand goes uncovered. For example, the customer arrives at the first node with 100% of its demand uncovered. If there is no branch opened at that node, entire demand remains to be uncovered and "survives" to go the next node. However, if a branch is opened at that location, depending on its size a certain proportion of the demand may be covered while the remaining proportion survives. The customer then goes to the second node, and the process is repeated, and then to the third, fourth, etc. When all the nodes are visited in this manner, we would have the actual proportion of uncovered demand for a given configuration of branch location and sizes.

Before the formal description of the linearization technique, we also like to demonstrate it on a small example with one demand node and three candidate branch locations. We further consider only one branch size option for the branches. For simplicity, we do not represent center locations because they have no impact on the coverage. Since we have a single demand node and a single branch size option, we simplify the notation by omitting i and m indices. In this example, the proportions of the demand that can be covered by branches at location $j \in \{1, 2, 3\}$ are given as $p_1 = 0.8$, $p_2 = 0.3$, and $p_3 = 0.6$. Suppose that in the solution, branches are opened at locations 1 and 3 but not at location 2 (i.e., $x_1 = x_3 = 1$ and $x_2 = 0$) yielding a total of $1 - (1 - 0.8)(1 - 0.6) = 0.92$ coverage for this demand node (see Fig. 2).

We now create a new network and define a few auxiliary variables that will enable us to accurately reach at this coverage value. First, we define a new set of parameters called the "scale factors", i.e., the proportion of demand *not* covered by a location depending on whether a branch is opened or not. For example, if a branch is opened at location 1, then the scale factor of this location is found as $q_1^+ = 1 - p_1 = 0.2$. However, if no branch is opened at this location then the scale factor is $q_1^- = 1$, i.e., no demand is covered. Similarly, we can set
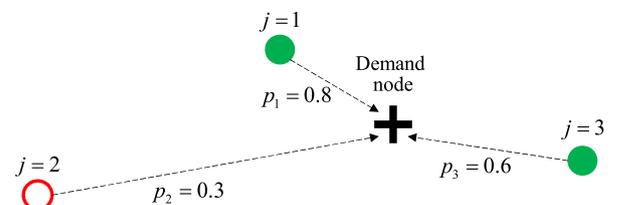
Fig. 2. An example problem with one demand node (denoted by a $+$ sign), and three candidate locations (denoted by circles). The two solid circles represent locations with open branches.
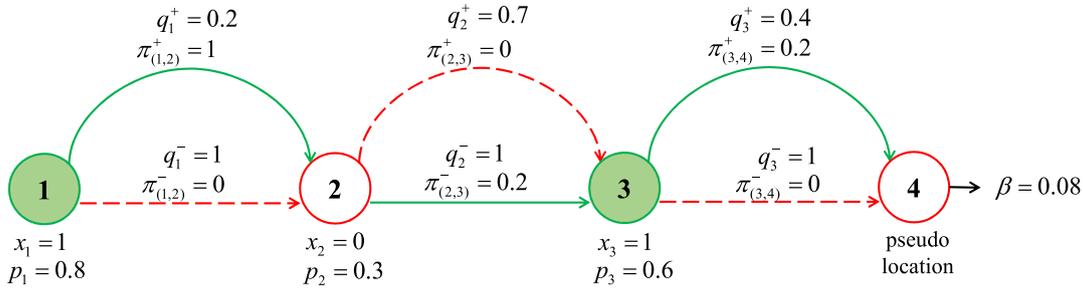
Fig. 3. Representation of the scaled transitory survival probability flows for an example problem. Positive and negative arcs are denoted by solid and dashed lines, respectively. Each disk represents a candidate location.

$q_2^+ = 1 - p_2 = 0.7$, $q_3^+ = 1 - p_3 = 0.4$, and $q_2^- = q_3^- = 1$. Next, we define a new set of variables called the "transitory survival probabilities" with negative and positive upper index to keep track of the overall proportion of customer demand that is *not* covered by the set of locations that are already "visited". The network in Fig. 3 depicts the values of transitory survival probabilities, scale factors, and the uncovered proportion of demand for this problem. In the figure, $\pi_{(1,2)}^+ + \pi_{(1,2)}^- = 1$ represents the transitory survival probability, i.e., proportion of demand *not* covered by any facility yet (even the first facility). After the first location is visited, the proportion of demand left uncovered is calculated as $q_1^+ \pi_{(1,2)}^+ + q_1^- \pi_{(1,2)}^- = 0.2 \times 1 + 1 \times 0 = 0.2$. That is, when a branch is opened at location 1, the overall proportion of demand not covered by this location is 0.2. The uncovered portion becomes the survival probability before the second location is visited, i.e., $\pi_{(2,3)}^+ + \pi_{(2,3)}^- = 0.2$. After the second location is considered, the survival probability becomes $q_2^+ \pi_{(2,3)}^+ + q_2^- \pi_{(2,3)}^- = 0.7 \times 0 + 1 \times 0.2 = 0.2$ again, which is expected because no branch is opened at location 2 and the same proportion continues to be uncovered. Finally, we visit the third location with $\pi_{(3,4)}^+ + \pi_{(3,4)}^- = 0.2$. After the third location, the total uncovered demand proportion is calculated as $\beta = q_3^- \pi_{(3,4)}^- + q_3^+ \pi_{(3,4)}^+ = 1 \times 0 + 0.2 \times 0.4 = 0.08$, which leads to a total coverage of $1 - \beta = 1 - 0.08 = 0.92$ for this demand node. In the figure, each arc is associated with two parameters, the scale factors and the transitory survival probabilities. Those arcs that are active are represented by solid lines whereas the others are represented by dashed lines. Note that the transitory survival probabilities may be nonzero only on the active arcs (more on this shortly). Finally, we place the locations in the order of the index in the network, but they could be placed in any order.

There are two key issues to be considered for this technique to produce the correct coverage proportions. First, there must be balance equations going from one node to the other. After any node, the uncovered demand proportion should be conveyed as it is or discounted properly depending on whether a branch is opened there. As we will see such balance equations can be written in a rather standard way, as they are utilized above in the example. Second, at most one of the arcs between two consecutive nodes can be active and the model should find the "correct" survival probabilities with negative or positive upper index. We utilize the binary location decision variables for this purpose;

for example, if we add a constraint $\pi_{(2,3)}^+ \leqslant x_2$, then we ensure that $\pi_{(2,3)}^+ = 0$ because $x_2 = 0$. Due to the coverage maximization objective such constraints together with the balance equations are sufficient to set these probabilities the correct values. Hence, if a branch is not opened at a location, the transition probability with negative upper index will convey all of the uncovered proportion to the next node. However, if a branch is opened there the transition probability with positive upper index convey it all and will be discounted with the scale factor for the next node. In the more general case when there are multiple size options, we will use a generalized version of these inequalities where there are multiple transition probabilities with positive upper index, one for each size option. In the more general case too, only one of those arcs will be active.

We now formally describe the technique. Consider a network composed of $|J| + 1$ nodes ordered consecutively and $|M_j|$ number of "positive-labelled" and one "negative-labeled" arcs between each neighbor node pair $(j, j + 1)$. Each node in the network denotes a candidate branch location $j \in J$ and the arcs represent the decisions of opening a branch of certain size or not opening at all at that node. Fig. 4 illustrates this network which represents a flow for a single demand node i. To implement the method described above to the more general case, we add the following decision variables:

$\pi_{i(j,j+1)m}^+$: transitory survival probability with positive upper index of demand node i between the jth and $(j + 1)^{th}$ candidate locations for size level m,

$\pi_{i(j,j+1)}^-$: transitory survival probability with negative upper index of demand node i between the jth and $(j + 1)^{th}$ candidate locations,

β_i: overall proportion of uncovered demand of demand node i. Note that using the definition of P_i given in Eq. (2), this probability is simply $\beta_i = 1 - P_i$, $\forall i \in I$.

We also define the following parameters:

q_{ijm}^+: Scale factor for the transitory survival probability of demand i if a branch of size m is installed at the jth candidate location, $q_{ijm}^+ = 1 - p_{ijm}$, $\forall i \in I, j \in J, m \in M_j$,

q_{ij}^-: Scale factor for the transitory survival probability of demand i if no branch is placed at the jth candidate location,
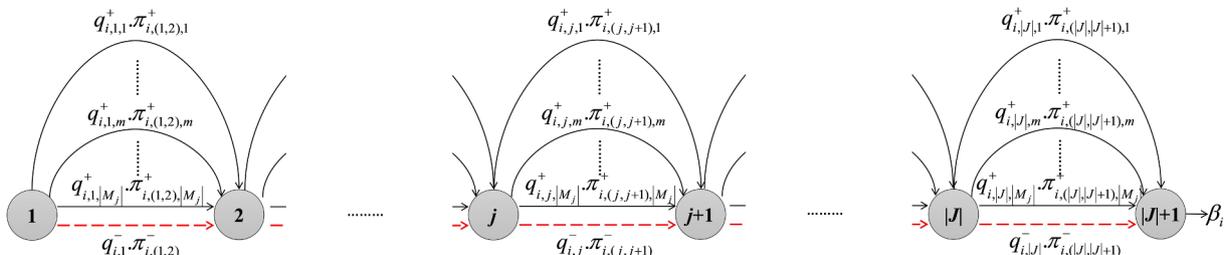
Fig. 4. Representation of the scaled transitory survival probability flow between candidate locations for a single demand node i. Positive and negative arcs are denoted by solid and dashed lines, respectively. Each disk represents a candidate location.

$$q_{ij}^- = 1, \ \forall \ i \in I, j \in J.$$

Note that these variables and parameters are generalized versions of those used in the example. Now they are defined for each demand node and extended for multiple size options. If $x_{jm} = 1$, then the survival probability with positive upper index leaving node j is scaled down by q_{ijm}^+. If the candidate location corresponding to a node j does not have a branch installed, the survival probability remains the same, i.e. scaled by $q_{ij}^- = 1, \ \forall \ i \in I, j \in J$. In our MILP formulation, we ensure that a transitory survival probability with positive upper index between consecutive nodes $(j, j + 1)$ gets a positive value only if a branch of size level m is installed at location j. On the other hand, the transitory survival probability with negative upper index gets a positive value if no branch is placed at j. Therefore, the total survival probability going from one node to the other either decreases or remains unchanged. In the MILP formulation, we compute the overall proportion of uncovered demand of a demand node i by the variable β_i. Note that node $|J| + 1$ is a pseudo location, it only represents the overall survival probability of a demand node i.

The final linearized formulation can be written as follows:

$$\max z_{MILP1} = \sum_{i \in I} \omega_i [1 - \beta_i] \tag{4a}$$

subject to
$$\pi_{i(1,2)}^- + \sum_{m \in M_j} \pi_{i(1,2)m}^+ = 1, \ \forall \ i \in I \tag{4b}$$

$$q_{ij}^- \pi_{i(j,j+1)}^- + \sum_{m \in M_j} q_{ijm}^+ \pi_{i(j,j+1)m}^+ = \pi_{i(j+1,j+2)}^- + \sum_{m \in M_j} \pi_{i(j+1,j+2)m}^+,$$
$$\forall \ i \in I, j \in \{1, 2, ..., |J| - 1\} \tag{4c}$$

$$q_{i,|J|}^- \pi_{i,(|J|,|J|+1)}^- + \sum_{m \in M_j} q_{i,|J|,m}^+ \pi_{i,(|J|,|J|+1),m}^+ = \beta_i, \ \forall \ i \in I \tag{4d}$$

$$\pi_{i(j,j+1)m}^+ \leqslant x_{jm}, \ \forall \ i \in I, j \in \{1, 2, ..., |J| - 1\}, m \in M_j \tag{4e}$$

$$\sum_{m \in M_j} x_{jm} \leqslant 1, \ \forall \ j \in J \tag{4f}$$

$$\sum_{n \in N_k} y_{kn} \leqslant 1, \ \forall \ k \in K \tag{4g}$$

$$z_{jk} \leqslant \sum_{m \in M_j} x_{jm}, \ \forall \ j \in J, k \in K_j \tag{4h}$$

$$z_{jk} \leqslant \sum_{n \in N_k} y_{kn}, \ \forall \ j \in J, k \in K_j \tag{4i}$$

$$\sum_{m \in M_j} x_{jm} = \sum_{k \in K_j} z_{jk}, \ \forall \ j \in J \tag{4j}$$

$$\sum_{n \in N_k} y_{kn} \leqslant \sum_{j \in J_k} z_{jk}, \ \forall \ k \in K \tag{4k}$$

$$\sum_{m \in M_j} x_{jm} \leqslant \sum_{k \in K_j} \sum_{n \in N_k} y_{kn}, \ \forall \ j \in J \tag{4l}$$

$$\sum_{j \in J} \sum_{m \in M_j} b_{jm} x_{jm} + \sum_{k \in K} \sum_{n \in N_k} c_{kn} y_{kn} + \sum_{j \in J} \sum_{k \in K} a_{jk} z_{jk} \leqslant B \tag{4m}$$

$$x_{jm} + z_{jk} - 1 \leqslant h_{jkm}, \ \forall \ j \in J, k \in K_j, m \in M_j \tag{4n}$$

$$\sum_{j \in J_k} \sum_{m \in M_j} \gamma_m h_{jkm} \leqslant \sum_{n \in N_k} Q_n y_{kn}, \ \forall \ k \in K \tag{4o}$$

$$x_{jm} \geqslant h_{jkm}, \ \forall \ j \in J, k \in K_j, m \in M_j \tag{4p}$$

$$z_{jk} \geqslant h_{jkm}, \ \forall \ j \in J, k \in K_j, m \in M_j \tag{4q}$$

$$x_{jm} \in \{0, 1\}, \ \forall \ j \in J, m \in M_j \tag{4r}$$

$$y_{kn} \in \{0, 1\}, \ \forall \ k \in K, n \in N_k \tag{4s}$$

$$z_{jk} \in \{0, 1\}, \ \forall \ j \in J, k \in K_j \tag{4t}$$

$$0 \leqslant h_{jkm} \leqslant 1, \ \forall \ j \in J, k \in K_j, m \in M_j \tag{4u}$$

$$\pi_{i(j,j+1)m}^+ \geqslant 0, \ \forall \ i \in I, j \in J, m \in M_j \tag{4v}$$

$$\pi_{i(j,j+1)}^- \geqslant 0, \ \forall \ i \in I, j \in J \tag{4w}$$

$$\beta_i \geqslant 0, \ \forall \ i \in I \tag{4x}$$

The objective function (4a) is equivalent to (3a) except it uses the survival probabilities β_i. Constraint set (4b) initializes the survival probability for each demand node i by setting their respective sums of transitory survival probabilities with negative and positive upper index to one. This is the first of the balance equations. Next, Constraint set (4c) presents the balance equations for the subsequent nodes, where total survival probability from one node to the other is adjusted for each demand node i using the corresponding scale factors. Finally, Constraint set (4d) calculates the overall coverage of demand node i by finding the survival probability from node $|J|$ to node $|J| + 1$. Constraint set (4e) ensures that the transitory survival probabilities with positive upper index are set to zero if no branch of the corresponding sizes are opened at location j. In essence this constraint ensures that at most one of the transitory probabilities may be positive. Constraint sets (4f)–(4m) are carried over from the nonlinear formulation unchanged and the Constraint sets (4n)–(4q) are for the other linearization as explained earlier. Finally, Constraint sets (4u)–(4x) declare variable definitions for the additional decision variables.

3.3. Alternative INLP and MILP formulations

Here we present an alternative formulation that follows a more direct approach. Instead of defining double-indexed variables for branch location and sizing and assignment to the centers, we define one triple-indexed variable to capture all those decisions. We continue to use the previous notation to a large extent, but we define a new binary variable which will allow us to discard the variables x_{jm} and z_{jk} used above. The new variable is defined as.

$$w_{jkm} = \begin{cases} 1, & \text{if a branch of size } m \in M_j \text{ is opened at location} \\ & \quad j \in J \text{ and connected to center } k \in K_j \\ 0, & \text{otherwise.} \end{cases}$$

The alternative formulation is:

$$\max z_{INLP2} = \sum_{i \in I} \omega_i \left[1 - \prod_{j \in J} \prod_{m \in M_j} (1 - p_{ijm} \sum_{k \in K_j} w_{jkm}) \right] \tag{5a}$$

subject to
$$\sum_{k \in K_j} \sum_{m \in M_j} w_{jkm} \leqslant 1, \ \forall \ j \in J \tag{5b}$$

$$\sum_{n \in N_k} y_{kn} \leqslant 1, \ \forall \ k \in K \tag{5c}$$

$$w_{jkm} \leqslant \sum_{n \in N_k} y_{kn}, \ \forall \ j \in J, k \in K_j, m \in M_j \tag{5d}$$

$$\sum_{j \in J} \sum_{m \in M_j} \sum_{k \in K_j} (a_{jk} + b_{jm}) w_{jkm} + \sum_{k \in K} \sum_{n \in N_k} c_{kn} y_{kn} \leqslant B \tag{5e}$$

$$\sum_{j \in J_k} \sum_{m \in M_j} \gamma_m w_{jkm} \leqslant \sum_{n \in N_k} Q_n y_{kn}, \ \forall \ k \in K \tag{5f}$$

$$w_{jkm} \in \{0, 1\}, \ \forall \ j \in J, k \in K_j, m \in M_j \tag{5g}$$

$$y_{kn} \in \{0, 1\}, \ \forall \ k \in K, n \in N_k \tag{5h}$$

Note that this formulation is equivalent to the previous one. First, one can recognize that the relationship $x_{jm} = \sum_{k \in K} w_{jkm}$ holds for each $j \in J$ and $m \in M_j$. The objective function (5a), the single-size restrictions for the branches and the centers (5b) and (5c), the budget constraint (5e) and the capacity constraints (5f) are immediate adaptations from the previous formulation. The new set of variables w_{jkm} captures the joint definitions of x_{jm} and z_{jk} and hence, eliminates the need for these variables as well as the constraints (3f) and (3g), which are used to identify branches that are connected to each center. As a result of this definition, now the capacity constraints (5f) are linear. The new set of variables is equivalent to the variables h_{jkm} that was used in the linearization of the previous formulation. Here, however, this new set of variables has to be defined as binary as opposed to continuous. Hence, in contrast to the previous formulation we have a larger number of binary variables, but may still reach at a stronger formulation that needs to be tested to be seen.

We now develop the mixed integer linear formulation by adapting the approach described in the previous section in a straightforward way. The only novelty here is the Constraint set (6e), which is an adaptation of the Constraint set (4e). The MILP is given as

$$\max z_{MILP2} = \sum_{i \in I} \omega_i [1 - \beta_i] \tag{6a}$$

$$\text{subject to} \quad \pi_{i(1,2)}^- + \sum_{m \in M_j} \pi_{i(1,2)m}^+ = 1, \ \forall i \in I \tag{6b}$$

$$q_{ij}^- \pi_{i(j,j+1)}^- + \sum_{m \in M_j} q_{ijm}^+ \pi_{i(j,j+1)m}^+ = \pi_{i(j+1,j+2)}^- + \sum_{m \in M_j} \pi_{i(j+1,j+2)m}^+,$$
$$\forall i \in I, j \in \{1, 2, ..., |J| - 1\} \tag{6c}$$

$$q_{i,|J|}^- \pi_{i(|J|,|J|+1)}^- + \sum_{m \in M_j} q_{i,|J|,m}^+ \pi_{i(|J|,|J|+1),m}^+ = \beta_i, \ \forall i \in I \tag{6d}$$

$$\pi_{i(j,j+1)m}^+ \leqslant \sum_{k \in K_j} w_{jkm}, \ \forall i \in I, \ j \in \{1, 2, ..., |J| - 1\}, \ m \in M_j \tag{6e}$$

$$\sum_{k \in K_j} \sum_{m \in M_j} w_{jkm} \leqslant 1, \ \forall j \in J \tag{6f}$$

$$\sum_{n \in N_k} y_{kn} \leqslant 1, \ \forall k \in K \tag{6g}$$

$$w_{jkm} \leqslant \sum_{n \in N_k} y_{kn}, \ \forall j \in J, k \in K_j, m \in M_j \tag{6h}$$

$$\sum_{n \in N_k} y_{kn} \leqslant \sum_{j \in J} \sum_{m \in M} w_{jkm}, \ \forall k \in K_j \tag{6i}$$

$$\sum_{j \in J} \sum_{m \in M_j} \sum_{k \in K_j} (a_{jk} + b_{jm}) w_{jkm} + \sum_{k \in K} \sum_{n \in N_k} c_{kn} y_{kn} \leqslant B \tag{6j}$$

$$\sum_{j \in J_k} \sum_{m \in M_j} \gamma_m w_{jkm} \leqslant \sum_{n \in N_k} Q_n y_{kn}, \ \forall k \in K \tag{6k}$$

$$w_{jmk} \in \{0, 1\}, \ \forall j \in J, m \in M_j, k \in K_j \tag{6l}$$

$$y_{kn} \in \{0, 1\}, \ \forall k \in K, n \in N_k \tag{6m}$$

$$\pi_{i(j,j+1)m}^+ \geqslant 0, \ \forall i \in I, j \in J, m \in M_j \tag{6n}$$

$$\pi_{i(j,j+1)}^- \geqslant 0, \ \forall i \in I, j \in J \tag{6o}$$

$$\beta_i \geqslant 0, \ \forall i \in I \tag{6p}$$

4. Numerical results and discussion

In this section we test the performance of the MILP formulations in terms of computation time and solution quality. For this purpose, we first generate three data sets each consisting of instances with different sizes. Each instance is then solved with the two MILP formulations for

optimality gaps of zero and 0.10. We denote our formulations as follows:

MILP1(0%) MILP formulation (4a)–(4x) as defined in Section 3.2 and solved with optimality gap zero,

MILP1(10%) MILP formulation (4a)–(4x) solved with optimality gap 0.10,

MILP2(0%) Alternative MILP reformulation (6a)–(6p) as defined in Section 3.3 and solved with optimality gap zero, and

MILP2(10%) Alternative MILP reformulation (6a)–(6p) solved with optimality gap 0.10.

4.1. Coverage function

There exist several empirical models in the literature to represent the gradual coverage. Some of them include exponential model (Chen, Li, He, Sun, & Chen, 2010; Yang et al., 2015; Karatas & Onggo, 2019), polynomial model (Liu & Towsley, 2004), Fermi-type model (Fewell & Ozols, 2011; Karatas, 2017), Elfes model (Kumar & Lobiyal, 2013), and cubic model (Xing et al., 2009). In this paper, we adopt the Fermi-type coverage function to serve as an example in the development of our numerical experiments, but our models can run on any model. Using the Fermi function, the coverage proportion of a demand node i by a branch of size m located at j is calculated as

$$f_m(d_{ij}) = \begin{cases} 1, & d_{ij} \leqslant \bar{d}_m, \\ \frac{1}{1 + 10^{[(d_{ij} - \bar{d}_m)/(\rho_m - \bar{d}_m) - 1]/\upsilon}}, & \bar{d}_m < d_{ij} \leqslant \bar{\bar{d}}_m, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

where υ is a "sensitivity" parameter that describes the "tailing" character of the function and ρ_m represents the range at which the coverage proportion is 0.5 for branch size m. Fig. 5(a) illustrates example coverage performances for four different size facilities computed by (1) with Fermi function parameters $\bar{d}_m \in \{1, 2, 5, 7\}$, $\bar{\bar{d}}_m \in \{8, 12, 16, 20\}$, $\rho_m \in \{6.5, 9, 12.5, 15.5\}$, and $\upsilon = 0.5$. Note that the coverage function is non-increasing in distance and non-decreasing in branch size level. Fig. 5(b) depicts an example of cooperative coverage of two branches computed by Eq. (2) and Fermi function (7).

4.2. Experiment settings

We benchmark the performance of all formulations on three data sets of different sizes, i.e. small, medium, and large. For each data set, we choose two alternatives for $|I|$ and $|J|$ and three alternatives for budget B. Next, we design a full factorial experiment with 12 combinations (also referred as "case") per data set. For each combination, we generate and solve 10 random instances, yielding a total of 360 instances. In each instance, we generate demand and candidate locations (for branches and centers) uniformly on a two-dimensional square region of size $s \times s$. We determine demand weights randomly from a uniform distribution between 0 and 1. The cost figures are also randomly drawn from various uniform distributions at each location, but the facility opening costs are scaled for different size options with fixed factors. Specific values of all sets of sizes, parameters, and coverage functions are summarized in Table 1. We determine subsets K_j and J_k by further restricting the possible center-branch assignments by imposing a distance threshold denoted by $d_{\min}$. We assume that an assignment is only possible if the distance between branch j and center k, d_{jk}, is less than $d_{\min}$. We formally define these two subsets as $K_j = \{k \in K | d_{jk} \leqslant d_{\min}\}, \forall j \in J$, and $J_k = \{j \in J | d_{jk} \leqslant d_{\min}\}, \forall k \in K$.

All models are implemented in General Algebraic Modeling System (GAMS©) and solved by GUROBI 5.0. The experiments are run on a PC with Intel(R) Core(TM) i7-8750H CPU@2.2 Ghz and 16 GB memory running on Microsoft Windows 10 64-Bit operating system. In all runs, we first "fix" the optimality gap to zero and maximum computational
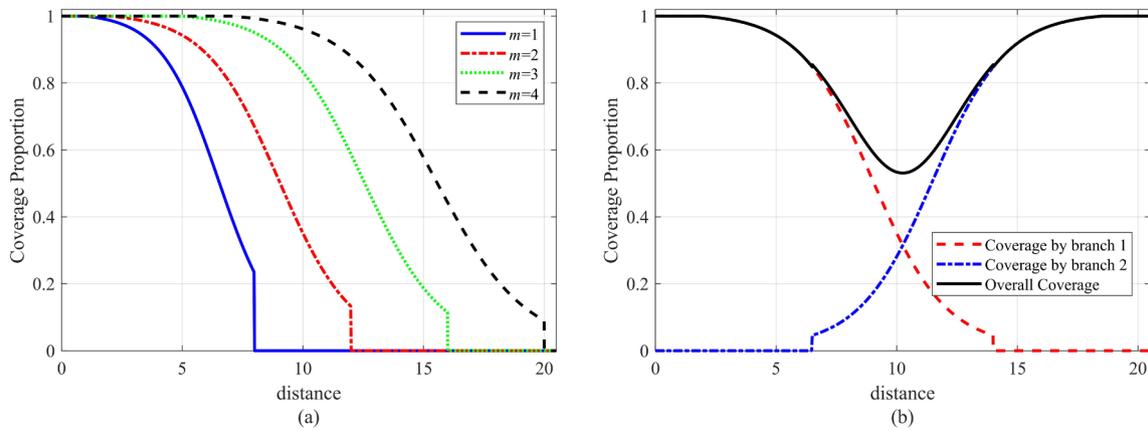
Fig. 5. (a) Coverage performance for 4 different size branches, (b) Individual and overall coverage performance of two branches.

time to one hour. It means that, a MILP run stops if a feasible solution is proven to be within 0.0% of the optimal objective value or the time limit is reached. We then solve MILP1 and MILP2 formulations for optimality gaps set to 0.10 while keeping the one hour time limit.

4.3. Results

Our experiments aim to compare the performance of all formulations proposed in this paper in terms of (i) the total number of instances solved to optimality, (ii) CPU times of the solved instances, and (iii) the optimality gaps of the unsolved instances. For each data set and problem size, Table 2 shows the total number of instances solved to

optimality within the one hour computing time limit, the average computing time (in seconds) among solved instances, and the average of the optimality gaps reported by the solver among unsolved instances when optimality gap is set to zero (for MILP1(0%) and MILP2(0%)), and all instances when optimality gap is set to 10% (for MILP1(10%) and MILP2(10%)). The row after each data set reports the overall average values for each performance metric. For each pair of MILP formulation (grouped with respect to the optimality gap setting) the largest number of instances solved, the smallest computing time, and the optimality gap are indicated in bold.

Table 1
Experiment setup for each instance set.

	Set 1: Small Cases	Set 2: Medium Cases	Set 3: Large Cases
Problem size	$|I| \in \{25, 50\}$ $|J| \in \{20, 40\}$ $|K| = 20$ $|M| = 2$ $|N| = 2$	$|I| \in \{100, 200\}$ $|J| \in \{50, 100\}$ $|K| = 50$ $|M| = 3$ $|N| = 3$	$|I| \in \{400, 800\}$ $|J| \in \{200, 400\}$ $|K| = 200$ $|M| = 4$ $|N| = 4$
Values of fixed parameters	$B \in \begin{cases} \{1000, 1500, 2000\}, |J| = 20 \\ \{2000, 2500, 3000\}, |J| = 40 \end{cases}$ $\gamma_m = \begin{cases} 100, m = 1 \\ 200, m = 2 \end{cases}$ $Q_n = \begin{cases} 200, n = 1 \\ 400, n = 2 \end{cases}$ $d_{min} = 20$ $s = 100$	$B \in \begin{cases} \{2000, 3500, 5000\}, |J| = 50 \\ \{3000, 5000, 7000\}, |J| = 100 \end{cases}$ $\gamma_m = \begin{cases} 100, m = 1 \\ 200, m = 2 \\ 400, m = 3 \end{cases}$ $Q_n = \begin{cases} 400, n = 1 \\ 800, n = 2 \\ 1200, n = 3 \end{cases}$ $d_{min} = 30$ $s = 125$	$B \in \begin{cases} \{5000, 7500, 10000\}, |J| = 200 \\ \{7500, 12500, 17500\}, |J| = 400 \end{cases}$ $\gamma_m = \begin{cases} 100, m = 1 \\ 200, m = 2 \\ 400, m = 3 \\ 800, m = 4 \end{cases}$ $Q_n = \begin{cases} 800, n = 1 \\ 1600, n = 2 \\ 3200, n = 3 \\ 6400, n = 4 \end{cases}$ $d_{min} = 40$ $s = 150$
Values of random parameters	$b_{jm} \in \begin{cases} U[30, 60], \forall j \in J, m = 1 \\ 1.75 \times b_{j1}, \forall j \in J, m = 2 \end{cases}$ $c_{kn} \in \begin{cases} U[50, 100], \forall k \in K, n = 1 \\ 1.5 \times c_{k1}, \forall k \in K, n = 2 \end{cases}$ $a_{jk} = 5d_{jk}, \forall j \in J, k \in K$ $\omega_i \in U[0, 1], \forall i \in I$	$b_{jm} \in \begin{cases} U[30, 60], \forall j \in J, m = 1 \\ 1.75 \times b_{j1}, \forall j \in J, m = 2 \\ 3 \times b_{j1}, \forall j \in J, m = 3 \end{cases}$ $c_{kn} \in \begin{cases} U[75, 125], \forall k \in K, n = 1 \\ 1.5 \times c_{k1}, \forall k \in K, n = 2 \\ 2 \times c_{k1}, \forall k \in K, n = 3 \end{cases}$ $a_{jk} = 5d_{jk}, \forall j \in J, k \in K$ $\omega_i \in U[0, 1], \forall i \in I$	$b_{jm} \in \begin{cases} U[30, 60], \forall j \in J, m = 1 \\ 1.75 \times b_{j1}, \forall j \in J, m = 2 \\ 3.2 \times b_{j1}, \forall j \in J, m = 3 \\ 4.4 \times b_{j1}, \forall j \in J, m = 4 \end{cases}$ $c_{kn} \in \begin{cases} U[100, 150], \forall k \in K, n = 1 \\ 1.5 \times c_{k1}, \forall k \in K, n = 2 \\ 2 \times c_{k1}, \forall k \in K, n = 3 \\ 2.75 \times c_{k1}, \forall k \in K, n = 4 \end{cases}$ $a_{jk} = 5d_{jk}, \forall j \in J, k \in K$ $\omega_i \in U[0, 1], \forall i \in I$
Coverage Function Parameters	$\bar{d}_m \in \{2, 3\}$ $\bar{\bar{d}}_m \in \{10, 12\}$ $\rho_m = (\bar{d}_m + \bar{\bar{d}}_m)/2$ $\upsilon = 0.5$	$\bar{d}_m \in \{2, 3, 4\}$ $\bar{\bar{d}}_m \in \{8, 10, 12\}$ $\rho_m = (\bar{d}_m + \bar{\bar{d}}_m)/2$ $\upsilon = 0.5$	$\bar{d}_m \in \{2, 3, 4, 5\}$ $\bar{\bar{d}}_m \in \{8, 10, 12, 14\}$ $\rho_m = (\bar{d}_m + \bar{\bar{d}}_m)/2$ $\upsilon = 0.5$

Table 2
Numerical results for all three data sets (10 instances per row).

Set #	Instance #	Problem size				# of solved instances				Computing time (s)*				Gap(%)**			
		$\|I\|$	$\|J\|$	$\|K\|$	B	MILP1(0%)	MILP2(0%)	MILP1(10%)	MILP2(10%)	MILP1(0%)	MILP2(0%)	MILP1(10%)	MILP2(10%)	MILP1(0%)	MILP2(0%)	MILP1(10%)	MILP2(10%)
Set 1 (small)	1-1	25	20	20	1000	10	10	10	10	**0.07**	**0.07**	**0.03**	**0.03**	-	-	**4.56%**	5.25%
	1-2	25	20	20	1500	10	10	10	10	0.07	**0.06**	**0.02**	**0.02**	-	-	4.17%	**3.77%**
	1-3	25	20	20	2000	10	10	10	10	0.04	**0.03**	**0.02**	**0.02**	-	-	2.00%	**0.75%**
	1-4	25	40	20	2000	10	10	10	10	**0.16**	0.18	0.05	**0.04**	-	-	4.34%	**3.44%**
	1-5	25	40	20	2500	10	10	10	10	**0.22**	0.26	0.05	**0.03**	-	-	**4.01%**	5.94%
	1-6	25	40	20	3000	10	10	10	10	0.27	**0.22**	0.04	**0.03**	-	-	4.84%	**2.83%**
	1-7	50	20	20	1000	10	10	10	10	0.07	**0.05**	**0.03**	**0.03**	-	-	**4.74%**	4.75%
	1-8	50	20	20	1500	10	10	10	10	0.10	**0.08**	**0.04**	**0.04**	-	-	4.95%	**4.82%**
	1-9	50	20	20	2000	10	10	10	10	0.07	**0.06**	0.03	**0.02**	-	-	**3.02%**	3.39%
	1-10	50	40	20	2000	10	10	10	10	0.26	**0.23**	0.08	**0.06**	-	-	5.30%	**5.01%**
	1-11	50	40	20	2500	10	10	10	10	0.29	**0.27**	**0.06**	**0.06**	-	-	6.65%	**5.38%**
	1-12	50	40	20	3000	10	10	10	10	0.26	**0.18**	0.06	**0.05**	-	-	6.47%	**4.98%**
	Average					10	10	10	10	0.16	0.14	0.04	0.04	-	-	4.59%	**4.19%**
Set 2 (medium)	2-1	100	50	50	2000	10	10	10	10	1.38	**1.10**	0.49	0.50	-	-	**3.44%**	4.32%
	2-2	100	50	50	3500	10	10	10	10	3.56	**1.33**	0.45	0.45	-	-	6.76%	**4.80%**
	2-3	100	50	50	5000	10	10	10	10	7.19	**4.07**	0.46	**0.44**	-	-	**5.84%**	6.99%
	2-4	100	100	50	3000	10	10	10	10	10.67	**2.88**	1.24	**1.15**	-	-	6.02%	**4.40%**
	2-5	100	100	50	5000	10	10	10	10	34.35	**14.91**	1.34	**1.15**	-	-	6.66%	**5.23%**
	2-6	100	100	50	7000	10	10	10	10	**43.62**	53.12	1.34	**1.21**	-	-	7.21%	**5.73%**
	2-7	200	50	50	2000	10	10	10	10	4.06	**2.36**	**0.55**	1.10	-	-	7.58%	**4.74%**
	2-8	200	50	50	3500	10	10	10	10	7.96	**2.78**	**0.68**	1.19	-	-	7.83%	**5.98%**
	2-9	200	50	50	5000	10	10	10	10	14.24	**3.71**	**0.77**	1.26	-	-	7.25%	**6.65%**
	2-10	200	100	50	3000	10	10	10	10	27.89	**8.74**	**2.20**	3.28	-	-	8.60%	**4.53%**
	2-11	200	100	50	5000	10	10	10	10	56.99	**15.97**	**2.56**	3.78	-	-	7.90%	**5.32%**
	2-12	200	100	50	7000	10	10	10	10	122.33	**80.87**	**3.02**	4.02	-	-	7.82%	**6.85%**
	Average					10	10	10	10	27.85	15.99	1.26	1.63	-	-	6.91%	**5.46%**
Set 3 (large)	3-1	400	200	200	5000	2	0	10	10	1243.14	-	144.98	**124.51**	3.71%	**2.77%**	9.40%	**8.71%**
	3-2	400	200	200	7500	0	0	10	10	-	-	**89.07**	143.20	3.89%	**3.19%**	9.45%	**8.54%**
	3-3	400	200	200	10000	0	0	10	10	-	-	**143.73**	183.63	4.09%	**3.75%**	9.07%	**8.21%**
	3-4	400	400	200	7500	0	0	10	10	-	-	1249.51	**1075.53**	**7.54%**	7.98%	**9.67%**	9.79%
	3-5	400	400	200	12500	0	0	4	**8**	-	-	2415.29	**2281.57**	10.65%	**9.51%**	10.98%	**10.00%**
	3-6	400	400	200	17500	0	0	2	0	-	-	1115.11	-	**11.12%**	11.20%	**11.18%**	11.20%
	3-7	800	200	200	5000	0	0	10	10	-	-	**171.88**	636.07	4.29%	**3.30%**	9.77%	**9.56%**
	3-8	800	200	200	7500	0	0	10	10	-	-	**240.47**	782.95	6.31%	**5.27%**	9.78%	**9.34%**
	3-9	800	200	200	10000	0	0	10	10	-	-	**664.19**	1044.24	7.25%	**6.69%**	9.92%	**9.81%**
	3-10	800	400	200	7500	0	0	2	2	-	-	2901.18	**2885.50**	11.45%	11.45%	11.47%	**11.46%**
	3-11	800	400	200	12500	0	0	0	0	-	-	-	-	**13.25%**	13.93%	**13.25%**	13.93%
	3-12	800	400	200	17500	0	0	0	0	-	-	-	-	**12.94%**	18.10%	**12.94%**	18.11%
	Average					0.17	0.00	6.50	6.67	1243.14	-	**913.54**	1017.47	8.04%	8.10%	**10.57%**	10.72%

∗ average among solved instances
∗∗ average among unsolved instances for MILP1(0%) and MILP2(0%), and all instances for MILP1(10%) and MILP2(10%)

4.3.1. Results for Set 1:

For Set 1, Table 2 shows that, both MILP1(0%) and MILP2(0%) are capable of solving all small size instances to optimality in less than a second. Similarly, MILP1(10%) and MILP2(10%) can also solve all instances in less than a second with average reported gaps of 4%. In this group of runs, the minimum and maximum gaps for MILP1(10%) are 2% (Case 1–3) and 6.65% (Case 1–11), and for MILP2(10%) these values are 0.75% (Case 1–3) and 5.94% (Case 1–5). We like to clarify that MILP1(10%) and MILP2(10%) stops the solution process as soon as the provable optimality gap drops to or below 10%, but for those runs we

reported the actual optimality gap, i.e., the gap with respect to the actual optimal solution (or the best upper bound found from any other formulation, when the optimal solution is not available). For this set of runs, we conclude that, both formulations have similar performances both in terms of solution time and quality.

4.3.2. Results for Set 2:

We now turn our attention to the results obtained on Set 2. Table 2 reveals that when optimality gap is set to zero, both MILP formulations can still solve all instances to optimality. On the other hand, having an

average CPU time of 28 s, MILP1(0%) is outperformed by MILP2(0%) which has an average CPU time of 16 s. Similarly, MILP1(10%) and MILP2(10%) can solve all instances within a few seconds.

Additionally, average gap for MILP2(10%) formulation is slightly better than that of MILP1(10%). Specifically, MILP1(10%) ends up with an approximate average gap of 7% (varying between 3.44% and 8.60%), while this value is approximately 5.5% (varying between 4.32% and 6.99%) for the MILP2(10%). Hence, it is fair to conclude that, in general, MILP2 is a slightly stronger formulation than MILP1 on these set of instances. At this point, we highlight the trade-off between the computational effort and solution quality. For example, in Instance 2–12, a decision maker can improve the computing time from 122 to 3 s for MILP1 or from 80 to 4 s for MILP2 at the cost of a 7% deterioration in solution quality.

4.3.3. Results for Set 3:

Regarding larger instances in Set 3, Table 2 reveals that none of the MILPs can be solved within one hour when optimality gap is set to zero except two instances out of 10 instances of Case 3–1 by MILP1. The average gaps are reported as 8.04% and 8.10% for MILP1(0%) and MILP2(0%), respectively. For MILP1(0%), the minimum and maximum average gaps are 3.71% (Case 3–1) and 13.25% (Case 3–11), and for MILP2(0%), these values are 2.77% (Case 3–1) and 18.10% (Case 3–12). When solving both MILPs with an optimality gap of 0.10, MILP2 performs slightly better in terms of number of instances solved, i.e. the total number of instances solved is 78 out of 120 for MILP1(10%) and 80 out of 120 for MILP2(10%). Among solved instances, MILP1(10%) is only slightly faster than MILP2(10%), with average computing times of 914 and 1017 s, respectively. In terms of the gaps reported by the solver, both MILP1(10%) and MILP2(10%) end up with an approximate average gap of 11%. For MILP1(10%), the minimum and maximum average gaps are 9.07% (Case 3–3) and 12.94% (Case 3–12), and for MILP2(10%), these values are 8.21% (Case 3–3) and 18.11% (Instance 3–12).

4.3.4. Summary

The performance of all formulations are presented in Figs. 6(a), (b), (c) for Sets 1, 2, and 3, respectively. The figures depict the percentage of instances solved within the one hour time limit (on the y-axis) within computing time (on the x-axis; logarithmic). Fig. 6(a) shows that MILP2(0%) performs consistently better than MILP1(0%) for the most part of the Set 1 runs. However, their rankings switch towards the right end of the figure. A similar behaviour is observed for MILP2(10%) and MILP1(10%). Note that, the differences in computing times for small size problems are very small and can be measured by milliseconds. In Fig. 6(b), we observe the dominance of MILP2(0%) over MILP1(0%) and MILP1(10%) over MILP2(10%). Specifically, both MILP1(0%) and MILP2(0%) are capable of solving all of the medium size instances. MILP1(10%) and MILP2(10%) can solve all instances in less than 4.23 and 5.11 s, respectively. Since none of the zero gap MILP2(0%) formulations were able to solve the large instances within the time limit, Fig. 6(c) only shows the performances of the MILP1(0%), MILP1(10%), and MILP2(10%). In this last figure, MILP1(10%) outperforms MILP2(10%) significantly.

Figs. 7(top) and (bottom) display the average and individual computing times of the MILP formulations for all instances. Solid lines with ● and ▲ represent average quantities over the 10 trials. ○ and △ represent CPU times for individual instances for zero and 0.10 gap settings and top bottom dotted lines represent the maximum and minimum CPU times observed among 10 instances for each case, respectively.

After these extensive numerical experiments on different size problem instances, we can conclude that MILP1 formulation leads to higher quality solutions and shorter computation times compared to MILP2 in general. To measure the effect of optimality gap on the performance of the formulations, we have also carried out all experiments with an optimality gap setting of 0.10. In those experiments we observed that computation times can be decreased significantly at the cost of a deterioration (approximately 3% for small, 4% for medium, and 10–14% for large instances) in the solution quality.

4.4. Solution characteristics

We now present some of the characteristics of the optimal (or near-optimal) solutions. Here, we would like to comment only on the results we obtained from MILP1(0%) as all the others show similar patterns. Some of these results are summarized in Table 3, which reports the average weighted demand coverage (objective function value z_{MILP1}), percentage of demand nodes not covered, partially covered, and fully covered, the number of branches and centers opened of each size, and the percentage of budget allocated to branches, centers, and assignments.

In each of the sets one notices that there are four subsets of cases classified according to their sizes. Each subset is further differentiated along three budget levels. As expected, the total coverage as well as the other measures of coverage (partially and full covered demand node percentages) all improve with the increasing budget. Similarly, the average number of branches opened also increases with the budget. We observe that there is a tendency to open more of the larger branches in small and medium cases, but the results are mixed for large instances. Fig. 8 (top) also reports on the average number of branches opened and the proportion of the total demand covered (another measure of coverage). The figure reports the average proportion of demand coverage as well as that of each instance. The small circles represent values obtained in individual replications, while red solid line depicts average values. Dashed lines represent the minimum and maximum demand coverage ratios observed among 10 replications for each case. From the figure, we can see that there is a large variation in our instances; some has total demand coverage of less than 10% (e.g., Case 1–7) while some others may have over 80% (e.g., Case 3–6). Similarly one can also see that there is a large variation on the number of branches opened in across different cases. On average, we can see that the number of branches opened is 10.4 for small instances, 20.3 for medium instances, and 46.4 for large instances. Recall that the number of alternatives for these sets are 20 and 40 for small, 50 and 100 for medium, and 200 and 400 for the large instances.

In our problem, the main function of centers is to provide the backbone capacity for the branches. Therefore, the number of open centers in our instances follow the pattern of branches to some extent, i.e., as the budgets increase so does the average number of centers opened. What we have observed differently, however, in our instances centers with the smaller sizes are generally opened more often than the large sizes (can also be observed in Fig. 8 (bottom)). This result might appear somewhat counter intuitive as the economies of scale in the center installation cost should favor opening larger centers rather than smaller ones. The main drivers of this result are the rather dispersed nature of branch locations and the presence of the assignment costs. In the cooperative coverage setting there is diminishing returns for each additional *nearby* branch opened (see Eq. (2)). Therefore, the model favors a more evenly dispersed branch location strategy to maximize the total coverage. To decide center locations to serve these branches, the model basically trade-offs center installation costs with the branch-center assignment costs. In the resolution of this trade-off it turns out that smaller centers are favored more than the larger ones. However, if the assignment costs get smaller we would observe that the model favors fewer but larger centers. In the extreme case where the assignment costs are negligible, the model would open the minimum number of centers needed to serve the branches. In terms of the budget use, although we see some variation, we observe that a major share goes to the branches, followed by centers and then by assignment costs. Roughly speaking, in all instances, about 40–50% of the budget goes to branch opening, about 25–35% goes to center opening, and about 20–30% goes to assignment costs.
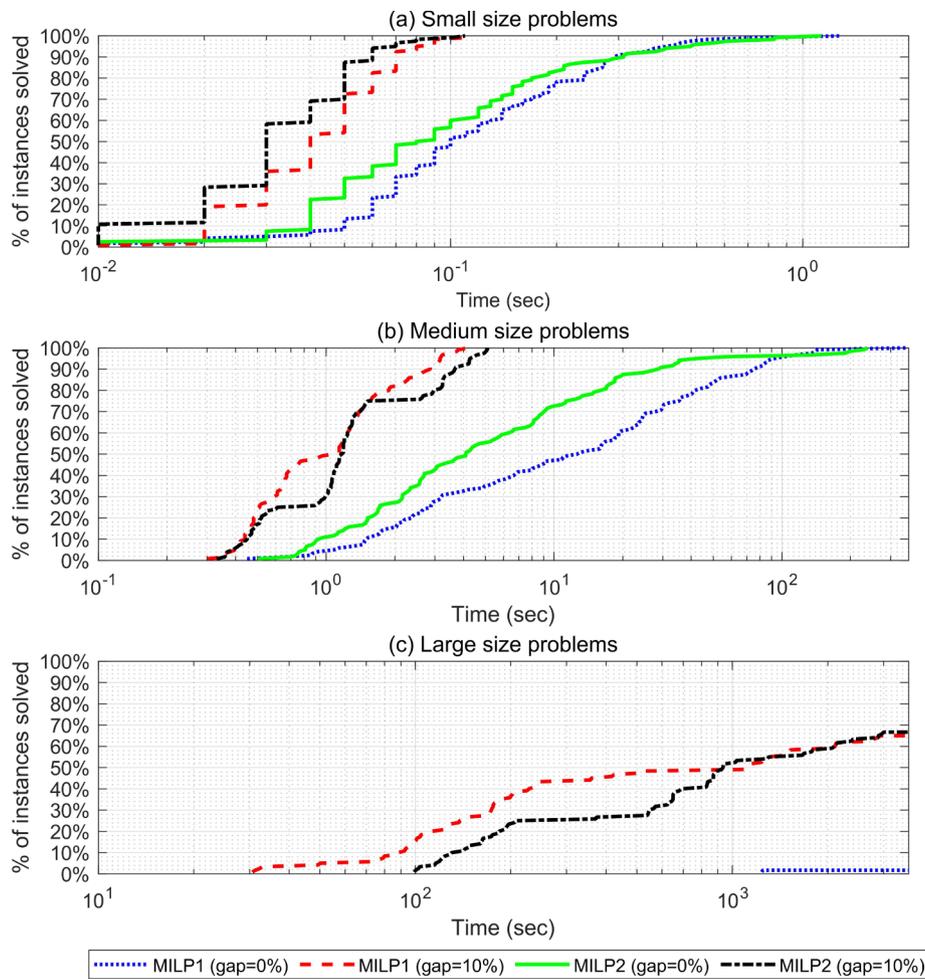
Fig. 6. Performance profile of each formulation for (a) Small size, (b) Medium size, and (c) Large size data sets.
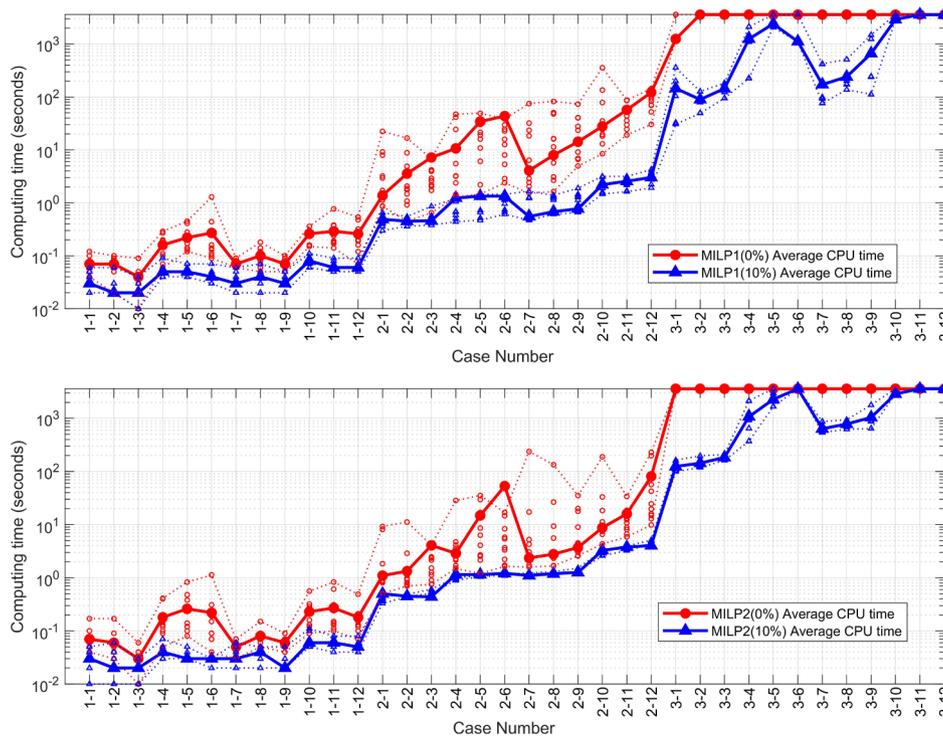
Fig. 7. Average and individual CPU times of the MILP1(0%) and MILP1(10%) [top] and MILP2(0%) and MILP2(10%) [bottom] for all test instances.

Table 3
Solution characteristics for MILP1(0%) (10 instances per row).

Set #	Instance #	Problem size						z_{MILP1}	% of demand			# of branches opened				# of centers opened				% of budget allocated to:		
		$|I|$	$|J|$	$|K|$	B	$|M|$	$|N|$		not covered	partially covered	fully covered	$m=1$	$m=2$	$m=3$	$m=4$	$n=1$	$n=2$	$n=3$	$n=4$	Branches	Centers	Assignments
Set 1 (small)	1-1	25	20	20	1000	2	2	3.39	63.20%	32.00%	4.80%	0.9	4.3	-	-	2.6	1.3	-	-	40.46%	33.87%	25.57%
	1-2	25	20	20	1500	2	2	2.81	61.20%	35.60%	3.20%	0.6	6.8	-	-	3.4	2.1	-	-	39.73%	32.45%	27.75%
	1-3	25	20	20	2000	2	2	2.89	59.20%	37.20%	3.60%	0.3	8.2	-	-	3.2	3.2	-	-	39.28%	34.40%	26.26%
	1-4	25	40	20	2000	2	2	5.21	40.80%	53.20%	6.00%	1.0	9.4	-	-	4.1	3.1	-	-	39.13%	30.93%	29.89%
	1-5	25	40	20	2500	2	2	5.50	34.40%	58.40%	7.20%	0.8	11.9	-	-	3.7	4.5	-	-	40.06%	30.48%	29.43%
	1-6	25	40	20	3000	2	2	6.02	29.20%	62.80%	8.00%	1.0	14.6	-	-	3.7	5.8	-	-	41.23%	30.36%	28.38%
	1-7	50	20	20	1000	2	2	4.51	73.00%	23.60%	3.40%	1.5	3.8	-	-	3.1	0.9	-	-	37.42%	32.92%	29.55%
	1-8	50	20	20	1500	2	2	6.39	62.00%	34.80%	3.20%	2.2	5.8	-	-	3.6	2.0	-	-	38.62%	31.26%	30.06%
	1-9	50	20	20	2000	2	2	7.66	52.80%	42.20%	5.00%	0.9	9.5	-	-	3.8	3.2	-	-	40.33%	31.57%	28.04%
	1-10	50	40	20	2000	2	2	9.94	46.60%	47.20%	6.20%	3.9	7.6	-	-	5.1	2.6	-	-	39.54%	31.71%	28.70%
	1-11	50	40	20	2500	2	2	10.06	42.40%	50.20%	7.40%	2.3	10.9	-	-	4.0	4.5	-	-	38.90%	31.42%	29.64%
	1-12	50	40	20	3000	2	2	10.45	37.40%	56.00%	6.60%	2.5	13.4	-	-	4.7	5.3	-	-	40.62%	30.56%	28.79%
	Average							6.24	50.18%	44.43%	5.38%	1.5	8.9	-	-	3.8	3.2	-	-	39.61%	31.83%	28.50%
Set 2 (medium)	2-1	100	50	50	2000	3	3	12.36	70.10%	25.40%	4.50%	2.5	3.6	3.6	-	5.7	0.6	0.1	-	44.35%	31.48%	24.12%
	2-2	100	50	50	3500	3	3	17.05	55.50%	39.40%	5.10%	3.1	4.5	8.6	-	7.0	2.1	0.6	-	46.15%	30.36%	23.47%
	2-3	100	50	50	5000	3	3	18.81	44.40%	48.90%	6.70%	2.9	3.5	15.0	-	4.9	4.4	1.8	-	48.44%	28.39%	23.14%
	2-4	100	100	50	3000	3	3	19.18	56.60%	37.30%	6.10%	5.9	5.9	4.2	-	6.5	1.7	0.1	-	41.63%	29.18%	29.15%
	2-5	100	100	50	5000	3	3	26.48	35.40%	55.00%	9.60%	6.8	7.3	10.4	-	7.3	4.4	0.5	-	44.75%	28.31%	26.92%
	2-6	100	100	50	7000	3	3	31.57	23.50%	63.10%	13.40%	7.2	6.0	18.9	-	5.0	5.8	3.1	-	47.75%	26.64%	25.60%
	2-7	200	50	50	2000	3	3	19.00	71.90%	23.70%	4.40%	2.0	4.0	4.1	-	5.0	1.1	0.1	-	44.51%	32.22%	23.23%
	2-8	200	50	50	3500	3	3	29.45	58.70%	35.10%	6.20%	2.8	5.3	8.5	-	6.5	2.5	0.4	-	46.51%	29.09%	24.36%
	2-9	200	50	50	5000	3	3	35.16	48.20%	44.65%	7.15%	2.5	5.3	14.2	-	5.6	4.8	1.2	-	48.23%	28.46%	23.29%
	2-10	200	100	50	3000	3	3	30.95	60.75%	33.80%	5.45%	4.7	5.9	5.2	-	5.7	2.3	0.0	-	43.51%	28.88%	27.58%
	2-11	200	100	50	5000	3	3	44.69	41.70%	49.35%	8.95%	5.3	8.9	11.1	-	8.0	3.1	1.3	-	46.76%	28.33%	24.89%
	2-12	200	100	50	7000	3	3	53.11	29.05%	60.50%	10.45%	7.0	10.9	16.2	-	7.0	6.0	2.1	-	47.47%	26.90%	25.61%
	Average							28.15	49.65%	43.02%	7.33%	4.4	5.9	10.0	-	6.2	3.2	0.9	-	45.84%	29.02%	25.11%
Set 3 (large)	3-1	400	200	200	5000	4	4	69.80	54.25%	38.80%	6.95%	4.6	8.6	6.6	4.4	10.2	1.2	0.0	0.0	49.58%	27.60%	22.80%
	3-2	400	200	200	7500	4	4	92.82	39.58%	50.20%	10.23%	6.0	13.4	6.7	9.1	12.9	2.4	0.4	0.0	50.75%	26.76%	22.48%
	3-3	400	200	200	10000	4	4	118.75	25.15%	60.40%	14.45%	9.4	14.2	10.4	12.6	14.8	3.8	0.8	0.0	52.02%	25.34%	22.64%
	3-4	400	400	200	7500	4	4	106.00	33.60%	55.00%	11.40%	9.8	14.8	7.6	6.6	15.2	1.6	0.0	0.0	48.49%	27.03%	24.46%
	3-5	400	400	200	12500	4	4	142.94	14.45%	68.60%	16.95%	17.9	19.9	10.3	14.6	24.7	1.1	1.9	0.2	49.51%	28.15%	22.34%
	3-6	400	400	200	17500	4	4	167.67	4.60%	72.90%	22.50%	22.6	20.8	11.8	25.6	33.8	3.0	2.2	1.0	50.15%	29.76%	20.08%
	3-7	800	200	200	5000	4	4	123.67	55.20%	37.25%	7.55%	3.8	6.2	4.2	8.2	9.2	2.4	0.0	0.0	51.88%	28.35%	19.75%
	3-8	800	200	200	7500	4	4	172.69	39.58%	50.43%	10.00%	3.8	7.4	7.0	13.8	13.2	3.2	0.4	0.0	52.88%	28.16%	18.94%
	3-9	800	200	200	10000	4	4	205.93	28.78%	59.33%	11.90%	5.6	14.6	9.0	15.4	14.6	4.0	1.4	0.0	53.13%	26.23%	20.63%
	3-10	800	400	200	7500	4	4	176.12	37.45%	52.18%	10.38%	5.6	12.0	8.0	10.2	16.8	1.2	0.2	0.2	50.31%	29.34%	20.34%
	3-11	800	400	200	12500	4	4	258.27	15.33%	68.20%	16.48%	10.2	15.2	9.0	22.0	22.2	1.2	3.2	0.2	50.68%	28.39%	20.92%
	3-12	800	400	200	17500	4	4	310.74	4.80%	74.93%	20.28%	12.8	20.6	13.0	31.2	21.6	2.0	5.2	2.0	52.02%	26.56%	21.40%
	Average							162.12	29.40%	57.35%	13.25%	9.3	14.0	8.6	14.5	17.4	2.3	1.3	0.3	50.95%	27.64%	21.40%

5. Conclusion

In this paper we present a two-level hierarchical facility location and sizing problem for maximizing expected proportion of demand covered. The lower level of facilities represent branches that act as first points of contact for the customers. The upper level facilities, on the other hand, represent centers that act as service suppliers to the branches at the lower level. The demand nodes are covered (satisfied) only by the lower level facilities and the proportion of a demand node covered depends on the proximity and the size of branches. We adopt the gradual and cooperative covering concepts from the location literature. We present two nonlinear formulations for the problem and develop their linearizations. Overall, our numerical experiments indicate that the MILP formulations constitute a powerful portfolio of location selection techniques, enabling the user to select the most appropriate balance of coverage maximization and computational speed
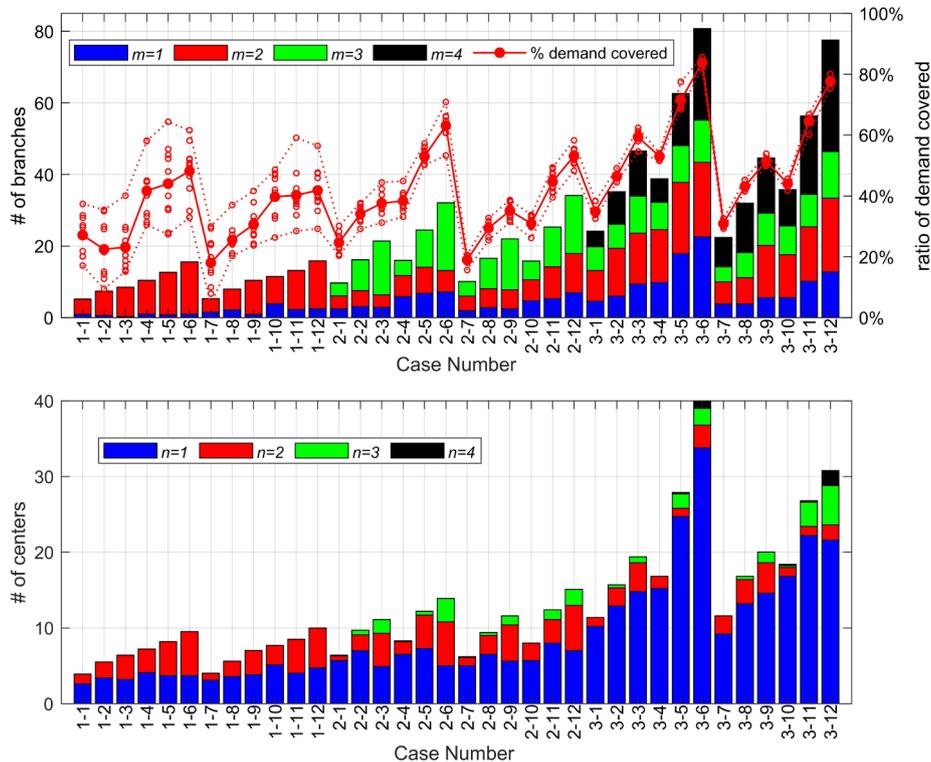
Fig. 8. Average number of branches opened and proportion of demand covered (top) and Average number of centers opened for each size for all cases (bottom).

for a variety of problems instances.

This work lends itself to a number of extensions. Firstly, the problem can benefit from specialized solution approaches. Although the commercial optimizers can solve some reasonably sized problems, if an application calls for larger sizes to be solved, they may be inadequate. Secondly, this work can be extended to include the presence of competing stores, which would necessitate a "capture" maximization of the problem. Finally, the model can also be extended to consider a profit maximizing objective also in the presence of important flow-based transportation costs. Such a formulation would require total demand covered at each specific branch and it would be a fruitful area to extend the research in that direction to better model many retail settings.

Declaration of Competing Interest

None.

Acknowledgements

None.

References

Afshari, H., & Peng, Q. (2014). Challenges and solutions for location of healthcare facilities. *Industrial Engineering and Management, 3*(1), 12.
Berman, O., Drezner, Z., & Krass, D. (2010). Generalized coverage: New developments in covering location models. *Computers & Operations Research, 37*(10), 1675–1687.
Berman, O., Drezner, Z., & Krass, D. (2018). The multiple gradual cover location problem. *Journal of the Operational Research Society,* 1–10.
Berman, O., Krass, D., & Drezner, Z. (2003). The gradual covering decay location problem on a network. *European Journal of Operational Research, 151*(3), 474–480.
Chan, Y., Mahan, J. M., Chrissis, J. W., Drake, D. A., & Wang, D. (2008). Hierarchical maximal-coverage location–allocation: Case of generalized search-and-rescue. *Computers & Operations Research, 35*(6), 1886–1904.
Chauhan, D., Unnikrishnan, A., & Figliozzi, M. (2019). Maximum coverage capacitated facility location problem with range constrained drones. *Transportation Research Part C: Emerging Technologies, 99*, 1–18.
Chen, J., Li, J., He, S., Sun, Y., & Chen, H.-H. (2010). Energy-efficient coverage based on probabilistic sensing model in wireless sensor networks. *IEEE Communications Letters, 14*(9), 833–835.
Farahani, R. Z., Hekmatfar, M., Fahimnia, B., & Kazemzadeh, N. (2014). Hierarchical facility location problem: Models, classifications, techniques, and applications. *Computers & Industrial Engineering, 68*, 104–117.
Fewell, M., Ozols, S., 2011. Simple detection-performance analysis of multistatic sonar for anti-submarine warfare. Tech. rep., Defence Science and Technology Organisation Edinburgh (Australia) Maritime Operations Div.
Geoffrion, A. M., & Graves, G. W. (1974). Multicommodity distribution system design by benders decomposition. *Management Science, 20*(5), 822–844.
Güneş, E.D., Nickel, S., 2015. Location problems in healthcare. In Location science. Springer (pp. 555–579).
Karatas, M. (2017). A multi-objective facility location problem in the presence of variable gradual coverage performance and cooperative cover. *European Journal of Operational Research, 262*(3), 1040–1051.
Karatas, M. (2018). Optimal deployment of heterogeneous sensor networks for a hybrid point and barrier coverage application. *Computer Networks, 132*, 129–144.
Karatas, M., & Onggo, B. S. (2019). Optimising the barrier coverage of a wireless sensor network with hub-and-spoke topology using mathematical and simulation models. *Computers & Operations Research, 106*, 36–48.
Kumar, S., & Lobiyal, D. (2013). Sensing coverage prediction for wireless sensor networks in shadowed and multipath environment. *The Scientific World Journal, 2013*.
Li, H., Zhao, L., Huang, R., & Hu, Q. (2017). Hierarchical earthquake shelter planning in urban areas: A case for Shanghai in China. *International Journal of Disaster Risk Reduction, 22*, 431–446.
Li, X., Ramshani, M., & Huang, Y. (2018). Cooperative maximal covering models for humanitarian relief chain management. *Computers & Industrial Engineering, 119*, 301–308.
Liu, B., & Towsley, D., 2004. A study of the coverage of large-scale sensor networks. In 2004 IEEE International conference on mobile ad-hoc and sensor systems (IEEE Cat. No. 04EX975). IEEE (pp. 475–483).
Min, H., & Melachrinoudis, E. (2001). The three-hierarchical location-allocation of banking facilities with risk and uncertainty. *International Transactions in Operational Research, 8*(4), 381–401.
Morton, D. P., Pan, F., & Saeger, K. J. (2007). Models for nuclear smuggling interdiction. *IIE Transactions, 39*(1), 3–14.
Narula, S. C. (1984). Hierarchical location-allocation problems: a classification scheme. *European Journal of Operational Research, 15*(1), 93–99.
Ortiz-Astorquiza, C., Contreras, I., & Laporte, G. (2018). Multi-level facility location problems. *European Journal of Operational Research, 267*(3), 791–805.
Paul, N. R., Lunday, B. J., & Nurre, S. G. (2017). A multiobjective, maximal conditional covering location problem applied to the relocation of hierarchical emergency response facilities. *Omega, 66*, 147–158.
Şahin, G., & Süral, H. (2007). A review of hierarchical facility location models. *Computers & Operations Research, 34*(8), 2310–2331.

Salmerón, J. (2012). Deception tactics for network interdiction: A multiobjective approach. *Networks, 60*(1), 45–58.

Skorin-Kapov, D., Skorin-Kapov, J., & Boljunčic, V., 2006. Location problems in telecommunications. In Handbook of optimization in telecommunications. Springer (pp. 517–544).

Teixeira, J. C., Bigotte, J. F., Repolho, H. M., & Antunes, A. P. (2019). Location of courts of justice: The making of the new judiciary map of Portugal. *European Journal of Operational Research, 272*(2), 608–620.

Xing, G., Tan, R., Liu, B., Wang, J., Jia, X., & Yi, C.-W. (2009). Data fusion improves the coverage of wireless sensor networks. *Proceedings of the 15th annual international conference on mobile computing and networking* (pp. 157–168). ACM.

Yang, Q., He, S., Li, J., Chen, J., & Sun, Y. (2015). Energy-efficient probabilistic area coverage in wireless sensor networks. *IEEE Transactions on Vehicular Technology, 64*(1), 367–377.