Accessible location of mobile labs for COVID-19 testing

Dianne Villicaña-Cervantes[1] · Omar J. Ibarra-Rojas[1]

Abstract

In this study, we address the problem of finding the best locations for mobile labs offering COVID-19 testing. We assume that people within known demand centroids have a degree of mobility, i.e., they can travel a reasonable distance, and mobile labs have a limited-and-variable service area. Thus, we define a location problem concerned with optimizing a measure representing the accessibility of service to its potential clients. In particular, we use the concepts of classical, gradual, and cooperative coverage to define a weighted sum of multiple accessibility indicators. We formulate our optimization problem via a mixed-integer linear program which is intractable by commercial solvers for large instances. In response, we designed a Biased Random-Key Genetic Algorithm to solve the defined problem; this is capable of obtaining high-quality feasible solutions over large numbers of instances in seconds. Moreover, we present insights derived from a case study into the locations of COVID-19 testing mobile laboratories in Nuevo Leon, Mexico. Our experimental results show that our optimization approach can be used as a diagnostic tool to determine the number of mobile labs needed to satisfy a set of demand centroids, assuming that users have reduced mobility due to the restrictions because of the pandemic.

Keywords Location · Accessibility · Mixed-integer linear program · BRKGA · COVID-19 testing · Operations research

1 Introduction

The COVID-19 pandemic, which began in 2019, was the cause of extreme precautionary measures worldwide to preserve the population's health. Some of the most common tools used have been testing strategies to identify areas with a large number of inhabitants infected with the SARS-CoV-2 (Severe Acute Respiratory Syndrome Coronavirus 2), as well as restricting the mobility of the population in order to foster social distancing (see [1, 2], and [3]). In this context, the government of the Nuevo Leon state of Mexico has a policy of supplying COVID-19 tests at no cost via official facilities (see [4]). Furthermore, one of the state's municipalities implemented the use of mobile laboratories for sample collection for COVID-19 tests (see [5]). Mobile laboratories consist of a vehicle and a medical crew along with testing equipment kept in a viable condition. Using

these, home sampling for COVID-19 tests is also possible since there is no need for sophisticated equipment to transport the test/samples. Another advantage of mobile labs is that they can be re-located according to the propagation of the infection around a specific region, i.e., the decision making can arise several times during a pandemic. Even given these clear benefits of using mobile labs, the planning process may be difficult since it should define a service network that is accessible for most of the region's people due to limited resources. In this study, we assume that it is necessary to support government decisions by locating mobile laboratories for COVID-19 testing and determining their service area, since the service area may depend on the number of medical staff or the number of vehicles available to meet the demand (see, for example, [6]). Our goal is to optimize the accessibility of the network of mobile labs to their potential user centroids; this is to foster low mobility among the population when seeking COVID-19 testing.

In general, reducing people's mobility was a global measure put in place to prevent COVID-19 infections. The main purpose behind improving accessibility is to provide good and diverse opportunities for satisfying a demand without the need for service users to travel long distances. However, since the concept of accessibility refers to the ability of a

✉ Omar J. Ibarra-Rojas
   omar.ibarrarj@uanl.edu.mx

   Dianne Villicaña-Cervantes
   dianne.villicanacts@uanl.edu.mx

1 Universidad Autónoma de Nuevo León, Av. Universidad s/n, San Nicolás de los Garza, Nuevo León, México

Fig. 1 Example of two mobile labs and two demand centroids
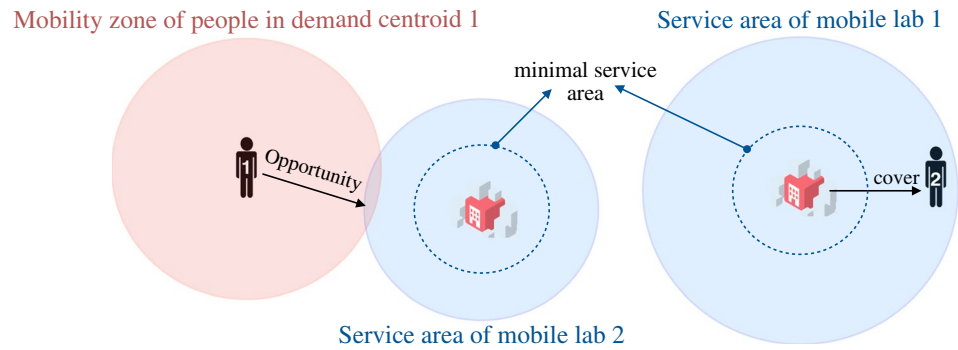
population in a region to reach a good or service, either commercial (restaurants, shops, etc.) or social (education and medical services), there is not one, unique and sufficient definition of the concept. The review by [7], concerning accessibility measures, states that a measure should include the cost incurred by individuals to cover the distance to the service, and the number and locations of opportunities, ideally in light of the spatial distribution of these opportunities. In particular, a function of travel cost models is that the greater the travel time or distance required to reach a destination, the less accessible that destination is. Similarly, [8] state that some important elements of an accessibility measure are the number of opportunities whereby clients can obtain a good or service, the mobility costs involved with satisfying the demand, and a spatial segregation measure which relates to the attempt to define an inclusive service network, i.e., to avoid cutting out regions of the potential service area (which is very important in the case of pandemics). Therefore, several indicators may be defined to represent important characteristics of an accessibility measure.

In response to these considerations, we aim to answer the following question: how do locate mobile labs for COVID-19 and determine their service radius to optimize the accessibility of the service network? The structure of our study is as follows. Section 2 addresses the problem statement, where we highlight the important elements to define accessibility. Section 3 presents the related literature, Section 4 introduces our mathematical modeling, Section 5 exhibits our solution approach, and Section 6 discusses our experimental results. Finally, conclusions and suggestions for further research are exhibited in Section 7.

2 Problem statement

This section presents a decision-making problem in designing a service network of mobile labs for COVID-19 testing. We assume sets of locations and demand centroids as given. Moreover, we know parameters for a mobility radius for inhabitants in each demand centroid, an initial service radius while defining a mobile lab, upper bounds for each

mobile lab's service radius, and the total service area among all labs. Then, our problem consists of determining the location of p mobile labs for COVID-19 testing and their service radius, optimizing the accessibility of the service network.

In particular, there are different essential elements to consider when defining accessibility, and those elements depend on the distance, the diversity, and the number of opportunities for users to reach the service. Figure 1 gives an example of two facilities along with their service areas (the dashed line denotes the minimal service area); two demand centroids are shown as well. Notice that users in demand centroids can be satisfied if they are within the service area of a currently open lab (for example, centroid 2 is covered by mobile lab1), or, alternatively, if the users can reach the service area of an open lab within their mobility area, which we define as "an opportunity" (see centroid 1). We highlight that demand satisfaction would not be possible in relation to this example network of labs if we considered only the minimal service areas for all the open labs. Moreover, the location and number of opportunities affect different characteristics of accessibility.

Based on the above, we use an accessibility measure similar to the one proposed by [9], which is defined as a weighted sum of the following accessibility indicators in Table 1, where we point out the relevance of each indicator.

Notice that we include the recommended elements of [7] and [8] to create an accessibility measure by using the above six indicators. Indeed, those indicators represent different characteristics of the accessibility of the service network for COVID-19 testing in relation to potential clients. As can be seen in the experimental stage, it is practically impossible to optimize all of the indicators simultaneously. In particular, Section 6 shows the levels reached by each one of the accessibility indicators in the course of our experimental work, and the preferences of the decision-maker can complement our optimization approach.

The main research contributions of our study are the following: a generalization of the problem proposed in [10], considering, in addition, variable radii and a re-definition of the accessibility indicators; a Biased random-key genetic

Table 1 Accessibility indicators to define our measure of accessibility to be optimized while designing the network of mobile labs for COVID-19 testing

Accessibility indicator	Relevance
Coverage: the number of demand centroids covered by the service areas of all the open labs (e.g., demand point 2 in Fig. 1 is covered by lab 1).	It may be desirable to increase the number of people served with no need for travel.
Minimum access: the number of demand centroids that are able to have their demand satisfied, either by simply being within the service area of an open lab, or by the fact that users in the demand centroid can reach (within their own mobility radius) the service area of an open facility.	Maximizing the latter indicator is essential to guarantee that the testing service is accessible for most of the region's population, even if it means that users must travel significant distances in order to access the testing process.
Mobility/transportation cost: for users not covered by the service area of any of the open labs, this indicator represents a "cost" (monetary, in time, in comfort, and so on) of traveling in order to have their demand satisfied.	Minimizing the mobility cost in the model lead to having labs near the potential users, as recommended by [7] and [8].
Proximity of the service: considering only the demand centroids that cannot be satisfied, this indicator represents the distances from each centroid to their closest open lab.	Users with no access to the system could be satisfied if the network is expanded in the future, e.g., by adding additional labs or additional transportation resources with social initiatives or by private companies.
Number of opportunities: the number of open labs with service areas intersecting the mobility zones of users in demand centroids not covered by the service area of open labs (for example, in Fig. 1, inhabitants in demand centroid 1, have an opportunity in lab 2).	Indeed, the number of opportunities is an important element in our context since people may look to schedule tests in a number of alternative labs, e.g., due to limited numbers of tests or to avoid crowded conditions.
Geographical segregation: This is related to the degree of dispersion (in terms of the distance) between the demand centroids with no access to the service.	As stated by [8], it is important to provide a good coverage of the region, avoiding leaving clusters of demand centroids that have no access to the service, or cut-off specific regions of the populations.

algorithm capable of obtaining high-quality feasible solutions for randomly-generated large instances in short computational times (commercial solvers that can not find even feasible solutions for such instances); and insights into a potential implementation taking account of the reduced mobility of users due to COVID-19 contingency measures.

3 Related literature

Facility location problems have been widely used to represent situations requiring decision-making processes involving the allocation of different types of health care facilities across multiple contexts. In this particular context, an important aspect of the situation is the accessibility of health services to potential clients; this can typically be represented based on a measure of the distance to travel to reach the service network (see, for example, [11]). However, as proposed by [12] and [13], other measures such as those which represent demand by potential clients with no access to the service are also necessary. Indeed, the recent study by [14] presents an exhaustive literature review of the topic of "Health Care Facility Location" (HCFL), emphasizing the difference between studies in the literature based on the objective to optimize (among other things), where these objectives are classified as follows: number of facilities, number of ambulances (transportation resources), travel time (or distance), costs, and direct-or-multiple coverage. In particular, these

reviewers present suggestions for further research directions, i.e.,'*relevant characteristics should be considered in location modeling to make location models more efficient*,". This is what we propose in this study by considering a general accessibility measure for the location of mobile labs in a pandemic. In addition, we focus on literature addressing the various different modeling approaches to the covering locations problem, with a particular interest in HCFL, because we use these approaches to define our optimization problem.

Classical coverage (as represented by our coverage and minimum access indicators) is defined in terms of a given service area of facilities and/or a base mobility distance for clients (see the review of [15]). It is an all-or-nothing measure, i.e., it classifies a demand point as covered if it is within the service area of open facilities. For example, the study of [12] addresses a multi-objective location problem of medical facilities, optimizing the distance from the demand points to the open facilities, the installation costs, and the number of potential users who are not covered by the service due to the limited capacity of facilities. Another example is the study of [16], which combines routing and maximal coverage locations for health centers. The objective was to minimize the weighted sum of transportation costs, the distances from demand points to their closest opportunity, and the fraction of uncovered demand when considering a predefined maximum distance. The authors implemented an ant-colony optimization and genetic algorithms to provide high-quality solutions for large instances. More recently, [17] presents a

literature review of the implementations of solutions to location problems with respect to humanitarian logistics, all in relation to practical cases where accessibility to humanitarian aid must be improved with regard to classical coverage (e.g., [18–21], and [22]).

In the context of cooperative coverage, it is considered that a demand point can be satisfied via the coverage of multiple facilities (as included in our accessibility indicator of the number of opportunities) or via an increase in resources. For example, [23] and [24] address the issue of double coverage as this occurs in a location problem concerning the location of ambulance stations in the context of health services. The authors optimize the weighted sum of the fractions of demand points satisfied by two ambulance stations and the installation costs; all this is subject to single and double coverage restrictions for some demand points. The authors of [25] also address double coverage by defining locations for air and ground emergency vehicles. The problem they moot defines a hard restriction for the coverage of all the demand centroids and its solutions are required to minimize operational costs. The study by [26] takes into account the circumstance that an emergency health center may be busy when receiving an aid call, and so the service may be forwarded to another center (this situation is termed inter-dependence). Then, it must be considered that clients are covered if they can be satisfied within a specific limited response time. The authors define a stochastic covering location problem based on emergency health centers which takes into account these inter-dependencies so that solutions must maximize the fraction of demand covered. They implement a commercial solver on a limited scenario to show the effectiveness of their proposed approach. The authors of [27] also model a stochastic location problem based on trauma centers and ambulance stations such that solutions must cover all demand points with minimal costs. The authors assume that patients can be transported to trauma centers either by ambulance or by helicopter.

Another interesting coverage concept is gradual coverage, where the level of satisfaction for a customer depends on the distance between the demand point and the service facility (similar to our mobility cost indicator); this contrasts with the classic all-or-nothing philosophy (see [28]). For example, the study by [29] investigates the locations of p ambulance centers; the authors maximize a measure of demand satisfaction in terms of two service radii. In particular, it is considered that there is complete coverage for a demand point if it is within a service radius $rmin$. That measure gradually decreases to 0 when the distance reaches $rmax$ (as between the client and the nearest ambulance center). The authors implement a particle swarm optimization algorithm to solve instances based on Malaysia's Johor Bahru region's data. The numerical results show a significant improvement in the quality of the solutions when considering larger service radii; this can be achieved by

making additional investments and/or selecting alternative transportation modes (as studied for supply chain cases in [30] and [31], respectively). Indeed, an essential characteristic of our optimization problem is the definition of variable service areas for facilities without ignoring other crucial elements as modeled using different coverage definitions.

In accordance with the general location theory, [32] states that the implementation of all the various different coverage concepts together can support decision-making. They integrated the gradual and cooperative coverage measures into a mathematical formulation that can be reduced to the classical version of gradual and cooperative coverage by setting some of its parameters. A similar example is the study by [33], which introduces a multi-objective approach which includes both gradual and cooperative coverage, in addition to considering variable service radii for facilities. The authors optimize a weighted sum of objectives based on the locations that are covered, the total cost, and the balance in terms of the service levels of open facilities. From a social perspective, the study by [10] implements a combination of classical, gradual, and cooperative coverage to define several accessibility indicators operative within a location problem which dictates that the solution recommends the opening of p facilities. Those authors proposed a mixed-integer linear formulation that can be solved with commercial solvers for instances representing up to 200 locations and 600 demand centroids. Similar to the study of [29], the authors in [10] also perform a sensitivity analysis demonstrating the potential benefit when the service area of facilities is increased. Based on all the above, we extend the integration of [10] to consider variable radii and implement the result in order to allocate mobile labs for COVID-19 testing, taking account of the problem that such testing implies the existence of restrictions leading to low-mobility conditions.

In summary, the studies in the literature show the implementation of various different coverage concepts related to health care facility location, and that the integration of these concepts can lead to improvements in the modeling of a service network (as it happens, in the commercial context). According to our literature review (see Table 2), our approach is one of the few that implements the several different definitions of coverage along with variable radii in order to define several accessibility indicators for health care services.

4 Mathematical modeling of the Accessibility Location Problem for COVID-19 testing

This section presents our mathematical representation of the proposed decision problem. First, we present the input data, and we recall the decision problem. Next, we introduce the

Table 2 Comparison of studies of Health Care Facility Location in the literature and our proposed approach

Study	Classic coverage coverage	Cooperative coverage coverage	Gradual coverage coverage	Variable radii	Objective
Stummer et al. [12]	✓				Proximity vs installation costs vs minimum access
Doerner et al. [16]	✓				Weighted sum of transportation costs, proximity and the fraction of uncovered demand
Abounacer et al. [18]	✓				Transportation time vs number of aiders needed vs coverage
Barzinpour & Esmaeili [19]	✓				Coverage vs operational costs
Jia et al. [20]	✓				Coverage
Murali et al. [21]	✓				Coverage
Salman & Yucel [22]	✓				Expected demand coverage
Gendreau et al. [23] and Doerner et al. [24]		✓			Weighted sum of minimum access and installation costs
Erdemir et al. [25]		✓			Operational costs
Grot et al. [26]		✓			Fraction of covered demand
Mousavi et al. [27]		✓			Expected operational cost
Azizan et al. [29]			✓		Demand satisfaction
Karatas [33]		✓	✓	✓	Weighted sum of coverage, costs, and balance of the level of service for installations
Ibarra-Rojas et al. [10]	✓	✓	✓		Weighted sum of coverage, minimum access, proximity, mobility costs, opportunities, and geographical segregation
Our study	✓	✓	✓	✓	Weighted sum of coverage, minimum access, proximity, mobility costs, opportunities, and geographical segregation

mathematical modeling of our accessibility measure and the accessibility indicators. Finally, we present the entire mathematical program of our optimization problem.

4.1 Input data and decision problem

The input of our problem is a set, I, of locations for mobile labs, and a number p, of labs that are to be opened at some locations in I. For each lab in a location i, its service radius must be at least a minimum of s_i, limited by a maximum increase of ε_i in that radius, as well as a total increase of l for all labs: i.e., the sum of all the increments of each lab is limited (as a budget). Let J be the set of demand centroids and let m_j be the mobility radius of the potential clients in demand point $j \in J$. We denote the distance from location $i \in I$ to the demand point $j \in J$ as d_{ij}. In addition, we define $A(j)$ as the set of all locations, i, that cover centroid j, assuming the maximum service radius, that is, $A(j) = \{i \in I : d_{ij} \leq s_i + \varepsilon_i + m_j\}$.

Based on the above data, our Accessibility Location Problem for COVID-19 testing (denoted as ALP) determines the location of p mobile labs and their service areas, maximizing a weighted sum of accessibility indicators (we detail the modeling in the following).

4.2 Modeling the accessibility measure

Note that a solution with p opened labs and service radii of ρ_i, leads to the sets $I_1 \subset I$ and $I_2 \subset I$ of open and closed labs, respectively. Similarly, there are sets $J_1 \subset J$ and $J_2 \subset J$ of demand centroids, covered and not covered by the service areas of labs in I_1, respectively. Figure 1 shows an example of three potential locations (denoted as rectangles) and six demand centroids (denoted by circles). The dashed lines around open labs $I_1 = \{1,3\}$ represent the minimal service radii, while the solid lines represent the service area ρ_i. Note that the demand centroid, 1, is covered by lab 1, while the demand centroid, 5, is covered by lab 3. The solid lines around the demand centroids which are not covered $J_2 = \{2,3,4,6\}$ represent their mobility radii. We say that a demand centroid j has an *opportunity* at an open lab at $i \in I_1$ if service area ρ_i intersects mobility radius m_j (users can reach the service area of the lab). Then, centroids 2 and 3 have an opportunity to be tested by lab 1, while demand centroids 4 and 2 have an opportunity to be tested by lab 3, and finally, demand centroid 6 has no such opportunities (Fig. 2).

Fig. 2 Example of a feasible solution for our decision problem
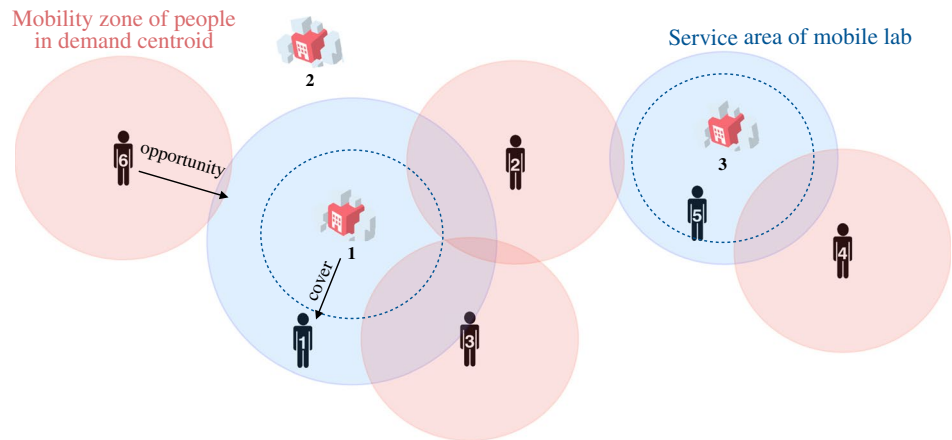
Now, we use the previous concepts to define our objective function. As we specify in Section 2, the accessibility measure is defined as a weighted sum of indicators. In particular, we will optimize the following function

$$\beta_1 \sum_{j \in J} \mathbf{a}_j + \beta_2 \sum_{j \in J} \mathbf{c}_j + \beta_3 \sum_{j \in J} \mathbf{t}_j + \beta_4 \sum_{j \in J} \mathbf{n}_j + \beta_5 \sum_{j \in J} \mathbf{o}_j + \beta_6 \mathbf{g},$$

where parameters β_k represent the weights of our accessibility indicators mentioned in Table 1 of Section 2. Indeed, we model these indicators in terms of classical, gradual, and cooperative coverage as follows.

– Coverage $\mathbf{c}_j$: binary indicator; this takes the value of 1 if the demand centroid j is covered by the service area of the network of mobile labs, and 0 otherwise (thus, it is a classical coverage indicator).

$$c_j = \begin{cases} 1 & \text{if } j \in J_1 \\ 0 & \text{otherwise} \end{cases}$$

– Minimum access $\mathbf{a}_j$: binary indicator that takes the value 0 if the inhabitants in the demand centroid j do not have access to the service, whereas it takes the value 1 if these clients can obtain service, either directly via the service area of an open lab, or because they have at least one opportunity. This may be seen as an all-or-nothing coverage indicator in terms of both service and mobility radius.

$$\mathbf{a}_j = \begin{cases} 1 & \text{if } j \in J_1, \text{or } d_{ij} \leq \rho_i + m_j \text{ for some } i \in I_1 \\ 0 & \text{otherwise} \end{cases}$$

– Mobility/transportation cost $\mathbf{t}_j$ often, the cost of travel from a point j to a point i, can be represented as a function of the distance, e.g., either proportional or inversely proportional to the distance depending on maximizing or minimizing while modeling (see [8]). In particular, we will use the function $f(d_{ij}) = \frac{1}{d_{ij}}$. Then, we define a gradual-cooperative coverage indicator $\mathbf{t}_j$

$\in (0,1)$ to represent mobility or transportation cost for inhabitants of demand point j. This indicator combines the inverses of the distances among all the potential opportunities in the set $A(j)$. Notice that by maximizing the value of $\mathbf{a}_j$, we guarantee the closest opportunities to the centroid j, and the indicator takes the value of 1 if all locations in $A(j)$ are used to open mobile labs.

$$\mathbf{t}_j = \begin{cases} 0 & \text{if } j \in J_1 \\ \dfrac{\sum\limits_{i \in A(j) \cap I_1} f(d_{ij})}{\sum\limits_{i \in A(j)} f(d_{ij})} & \text{if } j \in J_2 \end{cases}$$

– Proximity of the service $\mathbf{n}_j$: represents the travel distance from a point which is not covered, j, to the nearest open lab, which is minimized to bring mobile labs closer to potential clients.

$$\mathbf{n}_j = \begin{cases} 0 & \text{if } j \in J_1 \\ \{\min d_{ij} : i \in I_1\} & \text{if } j \in J_2 \end{cases}$$

– Number of opportunities $\mathbf{o}_j$: this represents the number of opportunities for potential clients in a demand centroid j not directly covered by the service area of any open lab.

$$\mathbf{o}_j = \begin{cases} 0 & \text{if } j \in J_1 \\ |A(j) \cap I_1| & \text{if } j \in J_2 \end{cases}$$

– Geographical segregation $\mathbf{g}$: represents the minimum distance between all pairs of the demand points which have no access to the service network of mobile labs. We maximize this last indicator to avoid the clustering of demand centroids which have no service.

$$\mathbf{g} = \min_{j,j' : \mathbf{a}_j = \mathbf{a}_{j'} = 0} \{d_{jj'}\}$$

The main difference between our work and that described in [10] is our definitions of coverage, and the fact that our opportunity indicators are based on the variable service radii of open mobile labs.

4.3 Mixed-integer linear program for the AFLP

To design a mathematical formulation for the ALP, we propose the following decision variables.

$$
z_i = \begin{cases} 1 & \text{if a mobile lab is opened at location } i \\ 0 & \text{otherwise;} \end{cases}
$$

x_i : increment of the service area of mobile lab in $i \in I$;

$$
u_{ij} = \begin{cases} 1 & \text{if demand centroid } j \text{ is covered by an open lab at } i \\ 0 & \text{otherwise;} \end{cases}
$$

$$
w_{ij} = \begin{cases} 1 & \text{if the mobile lab at } i \text{ is an opportunity for people in demand centroid } j \\ 0 & \text{otherwise;} \end{cases}
$$

$$
v_{ij} = \begin{cases} 1 & \text{if there is an open lab at } i, \text{ and it is the nearest one to demand centroid } j \\ 0 & \text{otherwise.} \end{cases}
$$

Moreover, variables $c_j \in \{0,1\}$, $a_j \in \{0,1\}$, $t_j \in [0,1]$, $n_j \geq 0$, $o_j \in \mathbb{N}$, and $g \geq 0$ represent the accessibility indicators described in Section 4.2.

We perform a normalization since we consider several terms in our accessibility measure. Hence, all terms are made to be values within the [0,1] interval, where 0 and 1 are the worst and best values for each indicator, respectively. In particular, indicators a_j, c_j, and t_j are already within [0,1]; thus, we divide them by $|J|$, so the sum of each indicator is also normalized. In the case of indicator n_j, we define an auxiliary term $\frac{\max_{i \in I}\{d_{ij}\} - n_j}{\max_{i \in I}\{d_{ij}\}} \in [0, 1]$, whose maximization is equivalent to the minimization of n_j. Moreover, we divide this term by $|J|$ so the sum is normalized. To achieve a normalization for the number of opportunities, o_j, we divide each indicator by the maximum number of opportunities, $|A(j)|$, and then divide by $|J|$. Finally, the indicator of geographical segregation, g, is divided by the largest distance among all pairs of demand centroids. Then, our mixed-integer linear program (MILP) for the ALP is defined as follows.

$$
\max \sum_{j \in J} \left(\beta_1 \frac{a_j}{|J|} + \beta_2 \frac{c_j}{|J|} + \beta_3 \frac{t_j}{|J|} + \beta_4 \frac{\frac{\max\{d_{ij}\} - n_j}{\max_{i \in I}\{d_{ij}\}}}{|J|} + \beta_5 \frac{\frac{o_j}{|A(j)|}}{|J|} \right) + \beta_6 \frac{g}{\max\{d_{ij}\}} \tag{1}
$$

subject to:

$$
\sum_{i \in I} z_i = p \tag{2}
$$

$$
\sum_{i \in I} x_i \leq l \tag{3}
$$

$$
x_i \leq \varepsilon_i z_i \qquad \forall i \in I \tag{4}
$$

$$
u_{ij} \leq z_i \qquad \forall i \in I, j \in J \tag{5}
$$

$$
(s_i + x_i) - d_{ij} \leq M(u_{ij} + 1 - z_i) \quad \forall i \in I, j \in J \tag{6}
$$

$$
d_{ij} - (s_i + x_i) \leq M(1 - u_{ij}) \qquad \forall i \in I, j \in J \tag{7}
$$

$$
w_{ij} \leq z_i \qquad \forall i \in I, j \in J \tag{8}
$$

$$
(s_i + x_i + m_j) - d_{ij} \leq M(w_{ij} + 1 - z_i) \quad \forall i \in I, j \in J \tag{9}
$$

$$
d_{ij} - (s_i + x_i + m_j) \leq M(1 - w_{ij}) \qquad \forall i \in I, j \in J \tag{10}
$$

$$
\sum_{i \in I} u_{ij} \leq M \cdot c_j \qquad \forall j \in J \tag{11}
$$

$$
c_j \leq \sum_{i \in I} u_{ij} \qquad \forall j \in J \tag{12}
$$

$$
v_{ij} \leq z_i \qquad \forall i \in I, j \in J \tag{13}
$$

$$
\sum_{i \in I} v_{ij} = (1 - c_j) \qquad \forall j \in J \tag{14}
$$

$$
a_j \leq c_j + \sum_{i \in I} w_{ij} \qquad \forall j \in J \tag{15}
$$

$$
t_j \leq (1 - c_j) \qquad \forall j \in J \tag{16}
$$

$$
t_j \leq \frac{\sum_{i \in I} \frac{w_{ij}}{d_{ij}}}{\sum_{i \in A(j)} \frac{1}{d_{ij}}} \qquad \forall j \in J \tag{17}
$$

$$
o_j \leq M(1 - c_j) \qquad \forall j \in J \tag{18}
$$

$$
o_j \leq \sum_{i \in I} w_{ij} \qquad \forall j \in J \tag{19}
$$

$$
n_j \leq \sum_{i \in I} v_{ij} d_{ij} \qquad \forall j \in J \tag{20}
$$

$$
g \leq d_{jj'} + M(a_j + a_{j'}) \qquad \forall j, j' \in J, j' \neq j \tag{21}
$$

$$
x_i \geq 0 \qquad \forall i \in I
$$
$$
z_i, u_{ij}, w_{ij}, v_{ij} \in \{0, 1\} \qquad \forall i \in I, j \in J
$$

The objective function (1) represents our accessibility measure, i.e., the weighted sum of the normalized indicators. Equation 2 guarantees the use of only p mobile labs, while inequality (3) bounds the total increment of the service area among all labs. Constraints (4), exist to avoid the situation whereby the maximal increment ε_i of the service area for mobile lab at i is surpassed. Inequalities (5), (6), and (7) ensure that a demand centroid j can be covered only by open labs with a service area greater or equal to the distance from i to j. Restrictions (8), (9), and (10) define an opportunity for demand centroid j, as an open lab at location i such that the service area of lab at i intersects the mobility zone of the population of j. Now, we define the constraints to model our accessibility indicators. First, inequalities (11) and (12) define the coverage of demand centroid j in terms of variables $\mathbf{u}_{ij}$. Then, Eqs. 13 and 14 define the nearest open lab for the demand centroids j which are not covered. Inequalities (15) allow the indicator $\mathbf{a}_j$ to be activated if demand centroid j is either covered ($\mathbf{c}_j = 1$), or clients in j have at least one opportunity. Constraints (16) and (17) define the value of indicator $\mathbf{t}_j$. Notice that by trying to maximize $\mathbf{t}_j$, we will be attempting to define close opportunities for centroid j; i.e., this activates variables $\mathbf{w}_{ij}$ for short distances d_{ij}. The number of opportunities $\mathbf{o}_j$ is defined through (18) and (19), and the distance from centroid j, which is not covered, to the nearest open lab is identified in Eq. 20. Finally, inequalities (21) determine the value of geographical segregation as the minimal distance among all pairs of demand centroids with no access to the service.

A preliminary experimental stage using a commercial solver found that it is possible to obtain near-optimal solutions in acceptable computational times for instances of up to 80 locations, $p = 10$ mobile labs, and 250 demand centroids. Nevertheless, in contrast to that yielded by the study by [10], which considers a fixed service area for each facility, the MILP yielded by our proposed AFL is intractable for large instances such as 200 locations and 600 demand centroids. This indicates that using variable service areas has a negative impact on the convergence of the commercial solver, and this is critical since we are dealing with the location of mobile labs where the decision making can arise several times during a pandemic. Moreover, a commercial solver can not find feasible solutions for large instances considering 400 demand points and 200 locations. In response, we designed and implemented a metaheuristic algorithm.

5 Biased Random-Key Genetic Algorithm for the ALP

We propose the implementation of a Biased Random-Key Genetic Algorithm (BRKGA) to obtain high-quality solutions for large instances in a reasonable length of time,

computationally. A BRKGA randomly generates a population of solutions, which will be combined and updated based on a fitness function until a stop criterion is met. The main characteristics of the BRKGA are the following: (i) a representation of solutions as random-keys, i.e., vectors of real numbers within [0, 1]; (ii) each random-key is associated with a feasible solution of the combinatorial optimization problem through a deterministic algorithm called the decoder; (iii) elitism is considered while choosing and combining "parent" solutions. The decoder algorithm is the only problem-dependent component of a BRKGA, and it maps the set of random-keys to the set of feasible solutions of the optimization problem and then evaluates them by returning the value of a fitness function for each. The latter characteristics lead to a search process that balances diversification and intensification (see [34]).

The BRKGA we use is illustrated in Fig. 3, where the combination of two solutions inherits each component of each of the two solutions according to a given probability. At the same time, we generate random vectors as mutants. Before the algorithm is run, values for the following parameters must be fixed: the number pop representing the population size; a scalar, pe, which is the fraction of solutions in the population that are to be considered as elite solutions, thus, $(1 - pe)100\%$ of the population are non-elite elements; a parameter pm representing the fraction of solutions in the population that will be replaced with mutants at each iteration; and finally, a probability ph of inheriting an element from the elite solution in the crossover.

As we have indicated, the decoder algorithm is the main component of the BRKGA. Hence, we describe below our decoder algorithm for the random-key (denoted as $\mathbf{r}$).

To design our decoder algorithm we first had to decide the size of the random-key $\mathbf{r}$ as $|I| + p$, i.e., $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_{|I|+p})$. The main idea is that the first $|I|$ elements in $\mathbf{r}$ will determine the locations at which facilities should be opened, and the last p elements are related to the service area of each open facility.

The decoder is detailed in Algorithm 1. The input is the random-key, the total increment available $l' = l$ of the service area for all mobile labs, the maximal increment ε_i of the service area for a mobile lab at location i, and the number p of mobile labs to be located. Each location, $i = 1, 2, \cdots, |I|$, is related with these positions in the random-key $\mathbf{r}$. Our decoder executes p main iterations. At each iteration k of our decoder, we open a mobile lab at the location i' (i.e., $\mathbf{z}_{i'} = 1$) with a maximum value of $\mathbf{r}_i : i = 1, 2, \cdots, |I|$, and we redefine $\mathbf{r}_{i'} = -1$ to avoid choosing the same location in following iterations. Moreover, the increment of the service area $\mathbf{x}_{i'}$ of the mobile lab at location i' is defined as the minimum between the available total increment l' and a fraction of the maximal increment $\varepsilon_{i'}$ in terms of element $\mathbf{r}_{|I|+k}$ of the random-key. In the end, we obtain a feasible solution to our

Fig. 3 Flowchart of the Biased Random-Key Genetic Algorithm (extracted from [35])
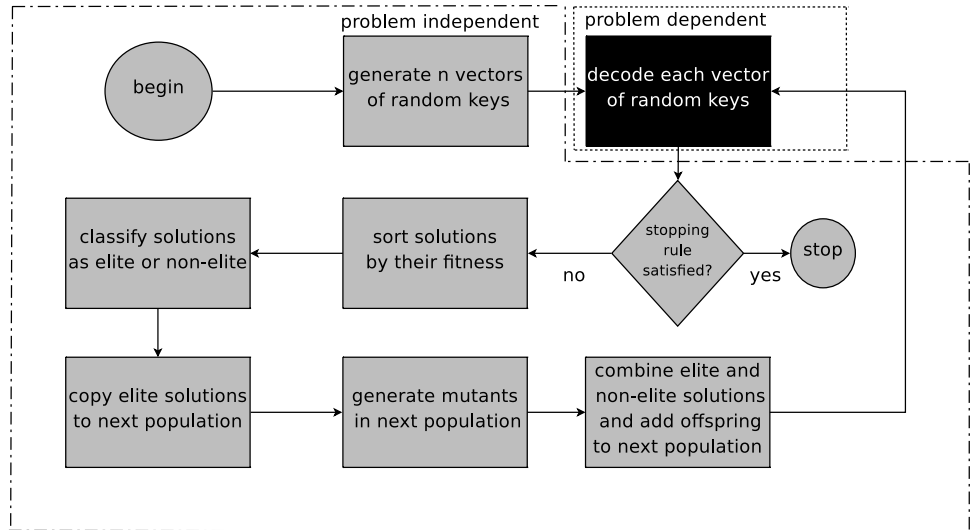
Algorithm 1 Decoder(r)

Input : $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_{|I|+p})$ and instance of AFL;
Initialize $l' = l$;
for $k = 1$ *to* p **do**
    $i' = \text{argmax}_{i \in I}\{\mathbf{r}_i\}$;
    Open mobile lab at i': $\mathbf{z}_{i'} = 1$, and update $\mathbf{r}_{i'} = -1$;
    Compute service area $\mathbf{x}_{i'} = \min\left(l', \lfloor \varepsilon_{i'} \cdot \mathbf{r}_{|I|+k} \rfloor\right)$;
    $l' = l' - \mathbf{x}_{i'}$;
Compute fitness function of feasible solution $(\mathbf{z}, \mathbf{x})$;

ALP. The fitness function calculates the value of our accessibility measure for the obtained solution.

We recall that constraints (5)–(21) model our accessibility indicators, while the feasibility of decisions for the location and increment of service radius is modeled by Eqs. 2-4. Notice that our Algorithm 1 chooses p locations, i.e., satisfies constraint (2) of $\sum_{i \in I} \mathbf{z}_i = p$. Moreover, at each iteration, the increment $\mathbf{x}_{i'}$ of the service area of the mobile lab at i' satisfies $\mathbf{x}_{i'} = \min\left(l', \lfloor \varepsilon_{i'} \cdot \mathbf{r}_{|I|+k} \rfloor\right) \leq l'$, and l' is updated as $l' = l' - \mathbf{x}_{i'}$, starting from a value of the total budget of increments of the service radius l; thus, we guarantee constraints (3) of $\sum_{i \in I} \mathbf{x}_i \leq l$. Finally, $\mathbf{x}_i = \min\left(l', \lfloor \varepsilon_{i'} \cdot \mathbf{r}_{|I|+k} \rfloor\right) \leq \varepsilon_i$ for all opened labs by the definition of our algorithm (constraints (4)). Therefore, we obtain a feasible solution for the AFL by implementing our decoder.

We show an example of our decoder for a scenario with six locations $|I| = 6$, a total increment of service radii of $l = 5$, an upper bound of the increment of the service radius for each mobile lab of $\varepsilon_i = 4$, and a number $p = 2$ of mobile labs to be located. Then, the size of the random-key is of $|I| + p = 8$ elements. For example, we take the vector $\mathbf{r} = [0.83 \,|\, 0.21 \,|\, 0.95 \,|\, 0.75 \,|\, 0.33 \,|\, 0.50 \,|\, 0.80 \,|\, 0.75]$, where elements from $\mathbf{r}_1$ to $\mathbf{r}_6$ will

define the location of the two mobile labs, and the service radii will take a value in terms of elements $\mathbf{r}_7$ and $\mathbf{r}_8$. In particular, the first iteration ($k = 1$) of our decoder defines the first location $i' = 3$ since $\mathbf{r}_3$ is the maximum value for the first six elements; thus, we open a mobile lab at $i' = 3$, and update the random-key as $\mathbf{r} = [0.83 \,|\, 0.21 \,|\, -1 \,|\, 0.75 \,|\, 0.33 \,|\, 0.50 \,|\, 0.80 \,|\, 0.75]$. Then, the increment of the service radius x_3 for mobile lab at location 3 is computed as $x_3 = \min(l', \lfloor \varepsilon_3 \cdot \mathbf{r}_7 \rfloor) = \min(5, \lfloor 4 \cdot 0.80 \rfloor) = 3$. Next, we update l' as $l' - x_3 = 2$. Similarly, the last iteration ($k = 2$) of the algorithm chooses location $i' = 1$ to open a mobile lab, and the increment of the service radius is computed as $x_1 = \min(l', \lfloor \varepsilon_1 \cdot \mathbf{r}_8 \rfloor) = \min(2, \lfloor 4 \cdot 0.75 \rfloor) = 2$. Finally, the algorithm determines the objective function for the generated solution.

Next, we demonstrate numerical results with respect to randomly-generated instances and with respect to a case study based on the location of mobile laboratories for COVID-19 testing in Nuevo Leon, Mexico.

6 Experimental results

6.1 Experimental environment and instances data set

Regarding the experimental environment, we coded the BRKGA using C++ and all the experiments were performed on a MacPro, 3.5 GHz 6-core Intel Xeon E5 with 16GB of RAM. Moreover, we used the solver, CPLEX 12.9 set with a stop criteria of 1% of relative gap[1] to obtain solutions for the

---

Table 3 Characteristics of instances types for the experimental stage

Instance type	$|I|$	p	$|J|$	s_i	ε_i	m_j	l	Grid size
A	40	5	120	2	5	15	8	100×100
B	80	10	250	2	5	15	15	130×130
C	200	25	600	2	5	15	38	150×150

MILP (1)–(21). The eight parameters of the objective function were fixed as proposed by [9], that is $\beta_1 = 0.5$, $\beta_2 = 0.25$, $\beta_3 = 0.125$, $\beta_4 = 0.0625$, $\beta_5 = 0.03125$, and $\beta_6 = 0.015625$. Finally, a preliminary experimental step was performed to calibrate the parameters of our BRKGA.

From that point on, we used the following values for the BRKGA's parameters: $pop = 500$, $pe = 0.20$, $pm = 0.10$, and $ph = 0.60$. The stop criterion was 100 iterations with no improvement.

We designed three instance types based on the cardinality of the set of locations for mobile labs ($|I|$), the number of demand centroids ($|J|$), and the number of facilities to be opened (p). In particular, we defined a grid to generate coordinates of locations and demand centroids randomly. We used the Euclidean distance for parameters d_{ij}. Table 3 details the characteristics of the instances types; we generated 30 instances of each type, leading to a total of 90 instances.

In fact, the size of the instances is related to the size of known real scenarios. In particular, we look at a real scenario that corresponds to "medium-size" instances (type B); then we generated smaller and larger values to analyze the potential implementation of our optimization approach across such differing scenarios.

Our experimental work focused on three steps. First, we compared our BRKGA set-up with running the CPLEX solver, limited to one hour of computational time (also, a stop criterion of 1% of the relative gap); this was to analyze the efficiency of our proposed approach. Then, we used the commercial solver but limited it to 10 minutes, in order to compare both methods under the assumption of limited time for decision-making; this is critical to cases such as humanitarian relief. Finally, we analyzed the solution of an instance based on the COVID-19 contingency measures in Nuevo Leon, Mexico.

6.2 BRKGA vs CPLEX limited to one hour of computational time

To compare the efficiency of the two algorithms, we focused on solution quality and computational time. Table 4 shows the average relative gap (*gap*) and the average computational time in seconds (*CPUtime*), as well as the standard deviation of both (*gap_stdev* and *time_stdev*, respectively), when considering 30 instances of each type. The first block of rows represent the results of applying the CPLEX solver limited

Table 4 Comparison of CPLEX limited to one hour of computational time and our proposed BRKGA

	Type A	Type B	Type C
CPLEX			
gap	**0.97**%	**4.20**%	32.83%
gap_stdev	0.04	4.62	4.08
CPUtime	10.26	2557.5703	3600
time_stdev	2.10	1125.64457	0
BRKGA			
gap	1.72%	5.82%	**32.38**%
gap_stdev	0.41	4.58	2.57
CPUtime	**4.52**	**23.04**	**197.13**
time_stdev	0.70	4.97	43.19

Bold entries are the best values of the average gap and the average time for each instance type

to one hour of computational time, and the second block of rows are the results of using our BRKGA.

Notice that we highlight with bold font the best values of the average gap and the average time for each instance type. For example, the CPLEX solver obtains better solutions on average for instances A and B, while it can be seen that solutions of instances C are slightly better when using BRKGA. In general, our BRKGA obtains solutions with a difference of less than 1.65%, for the relative gap, compared to the ones obtained by CPLEX. Notice that the value of the standard deviation of the relative gaps on instances of type B is significant. Moreover, we obtain a large average relative gap for instances C (more than 32% for both methods), which may indicate poor-quality dual bounds, and not necessarily low-quality solutions. The latter results also open the door to further research on how to improve the quality of the upper bounds definition: e.g., by designing and implementing valid inequalities.

In the case of computational times, the BRKGA uses less than 4 minutes for each of the instances, only a fraction of the time required by the commercial solver. In particular, the CPLEX solver reaches the time limit for all instances of type C and for some instances of type B (notice the large value of the standard deviation of the computational times). Only instances of type A are tractable by CPLEX in a matter of seconds. Based on the above results regarding both quality and computational time, it can be seen that our proposed BRKGA is a more efficient tool for helping in the decision-making process, as compared to a commercial solver.

To present a visual comparison, Fig. 4 shows the value of the accessibility measure for all the solutions obtained

Fig. 4 Value of the objective
function for all the feasible
solutions obtained by CPLEX
limited to one hour of com-
putational time and also those
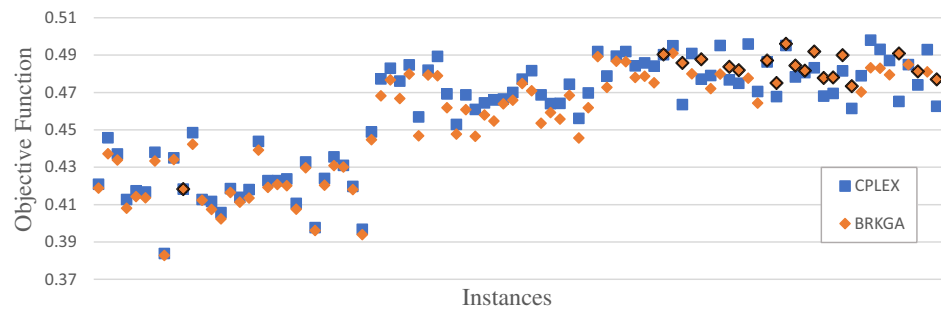yielded by our proposed
BRKGA

Table 5 The relative improvement of accessibility indicators achieved by implementing our BRKGA as compared to using the CPLEX solver

	a	c	t	n	o
A	0.05%	− 5.69%	1.78%	− 0.04%	0.85%
B	− 0.82%	− 5.91%	1.76%	1.07%	0.56%
C	1.16%	2.48%	2.65%	− 0.14%	2.61%

by CPLEX (denoted with squares) and for all the solutions yielded by our BRKGA (denoted with diamonds). We highlight with a borderline the solutions of BRKGA with a better value of the objective function as compared to the one obtained by CPLEX; this happens mainly in instances type C.

The objective function returns a single value within the interval [0,1], but it depends on several indicators. In response, we report a comparison of each term in the objective function, without considering the weight parameters. Table 5 shows the average relative improvement of each indicator for the solution obtained by our BRKGA as opposed to the one reported by CPLEX. The three rows represent the results for the three instance types. The columns represent the aggregated values for each indicator, which are denoted as **a**, **c**, **t**, **n**, and **o**. We did not report a comparison for indicator **g** since it is not defined in solutions with less than two demand centroids with no access to the service.

It should be noted that the solutions yielded by BRKGA obtain better values for mobility- or transportation costs (column **t**) and the number of opportunities (column **o**) for all instance types. In contrast, the solutions yielded by CPLEX have better values for the indicators of classic coverage (column **c**) and distance to the service network (column **n**), for two of the three instance types. Figure 5 details the relative improvements reported in Table 5, but for each

instance. Moreover, we report the absolute improvement for indicator **g** of geographical segregation for all instances.

Notice that indicators **t** and **n** have better values in solutions obtained by our BRKGA, mainly for instances of types B and C. Indicators **a** and **c** have better values in solutions obtained by CPLEX, mainly for instances of type B. In the case of indicator **o**, the more significant improvements achieved by BRKGA compared to CPLEX occur for instances of type C. Finally, values of indicator **g** are similar in both kinds of solution.

Here, we highlight the fact that we are not interested in analyzing the trade-offs between the different indicators; instead, we aim to provide a single solution to a location problem that can support the design of a service network for COVID-19 testing in a given region. Indeed, our study can be complemented by multicriteria decision methods to determine differing weight parameters based on subjective preferences in the decision-making process. In fact, we propose such studies as further research since this is out of the scope of the current project. Moreover, since we are dealing with the problem of allocating mobile labs, decision-makers may have to apply our approach at different instants of time along the timeline of a pandemic, and thus computational times may well become critical. We perform a similar comparison, but we use CPLEX limited to 10 minutes of computational time.

6.3 BRKGA vs CPLEX limited to 10 minutes

In some decision-making scenarios, time is a critical element. For example, humanitarian relief often requires fast decision-making in order to reduce and limit in time people's deprivation of essential items or services, and we are interested in the allocation of mobile COVID-19 testing labs, which can be reallocated more than once during a pandemic. Therefore, fast tools for decision-making are important. Table 6 exhibits the results of using both our BRKGA
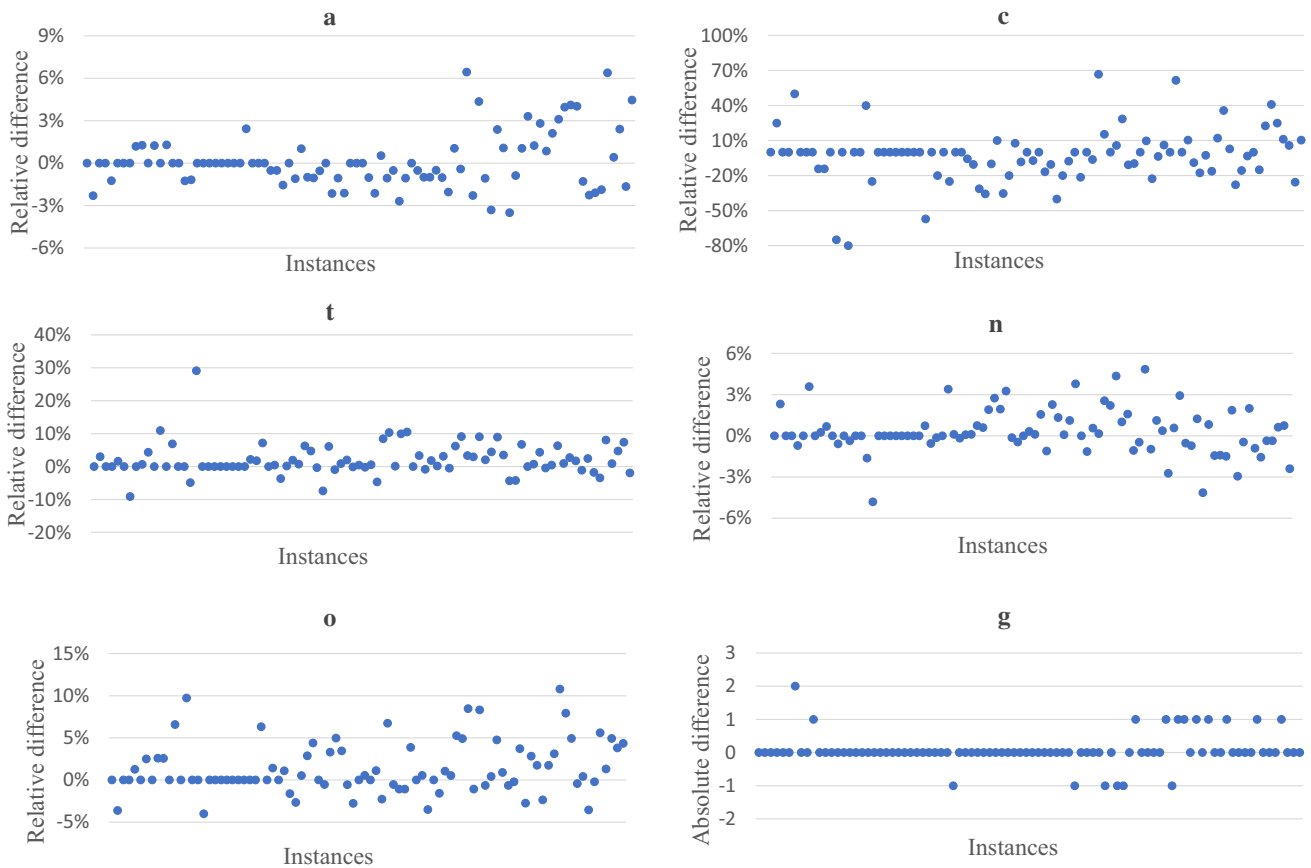
Fig. 5 Comparison of accessibility indicators for all the solutions obtained by CPLEX limited to one hour of computational time and our BRKGA

Table 6 Comparison of CPLEX limited to 10 minutes of computational time and our proposed BRKGA

	Type A	Type B	Type C
CPLEX			
gap	**0.97**%	**5.22**%	89.58%
gap_stdev	0.04	4.81	22.51
CPUtime	10.79	589	600.26
time_stdev	2.16	49.23	0.24
BRKGA			
gap	1.72%	5.82%	**32.38**%
gap_stdev	0.41	4.58	2.57
CPUtime	**4.52**	**23.04**	**197.13**
time_stdev	0.70	4.97	43.19

Bold entries are the best values of the average gap and the average time for each instance type

and the CPLEX solver with a stop criterion of 10 minutes (besides the stop criterion of a relative gap of 1%). In particular, the relative gap was computed using the dual bound found by applying the CPLEX solver limited to one hour of computational time.

Under the assumption of a small amount of time being available for the decision-making process, we can see that our BRKGA significantly outperforms CPLEX, for instances of type C, since we obtain an average relative gap of 89.58% in comparison to solutions obtained by CPLEX. Figure 6 shows the values of the accessibility measure for all the instances, where we highlight the large differences in the quality of the solutions.

Finally, Fig. 7 presents the relative improvement in indicators **a**, **c**, **t**, **n**, and **o** attained by the solutions yielded by our BRKGA, as compared to the ones reported by CPLEX. Since it is impossible to compute indicator **g** for a solution with less than two demand centroids with no access to the service, in this case, we compute the absolute difference between the value found by BRKGA and the one obtained by CPLEX.

Notice that the larger the instances, the larger the differences between the accessibility indicators in the solutions found by our BRKGA as compared to the commercial solver. In general, our proposed metaheuristic algorithm outperforms CPLEX for indicators of minimum access (**a**), classical coverage (**c**), mobility cost (**t**), and number of

Fig. 6 Value of the objective function for all the feasible solutions obtained by CPLEX and our ptoposed BRKGA, both limited to 10 minutes of computational time
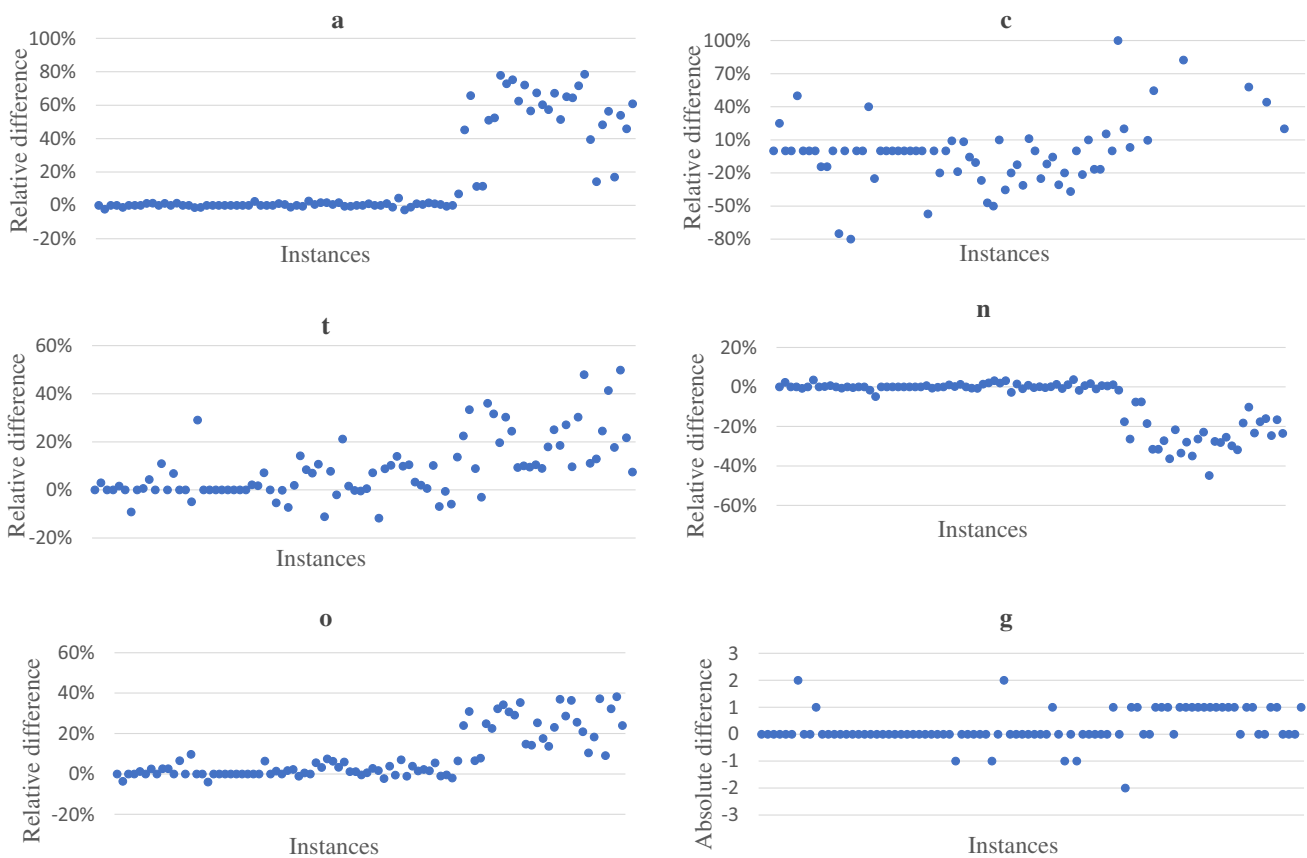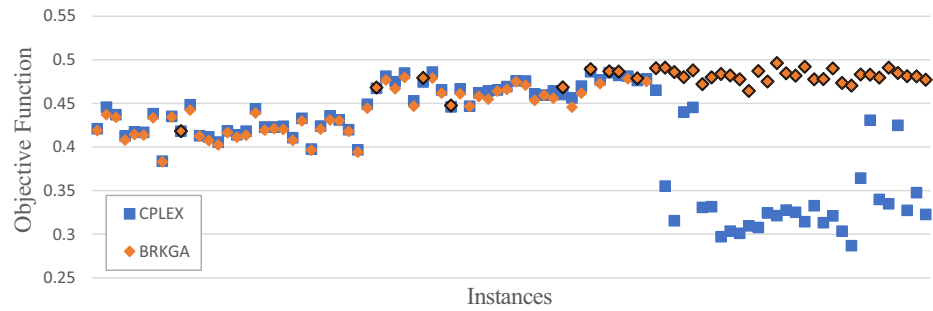
Fig. 7 Comparison of accessibility indicators for all the solutions obtained by CPLEX and our BRKGA, limited to 10 minutes of computational time

opportunities (**o**). However, we obtain comparable values of geographical segregation **g**, using both solution algorithms, and better results for the indicator of the distance to the service network (**n**) for large instances. The latter results show that the CPLEX solver, limited to short computational times, can generate solutions that include

shorter distances from demand centroids with no access to the service network, as compared to BRKGA.

Besides comparing CPLEX and our BRKGA, we performed numerical experimentation whereby we verified that the solutions obtained by our BRKGA satisfy local optimality for some neighborhood

Fig. 8 Comparison of accessibility indicators for the solutions obtained by the GA and BRKGA for the instances type D
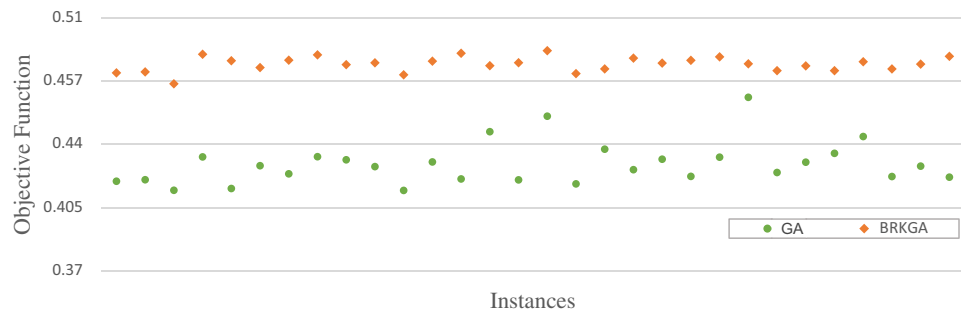
structures. In particular, we defined an operator $2\text{-}opt((i, i') : \mathbf{z}_i = 1, \mathbf{z}_{i'} = 0)$ which closes mobile lab at i, and opens a new lab at i'. Next, the increment of the service area of the mobile lab at i' is computed as $\mathbf{x}_{i'} = \min\{\mathbf{x}_i, \varepsilon_{i'}\}$, and hence, we can update the available increment for all mobile labs. We designed a local search based on our $2\text{-}opt()$ operator, which chooses the neighbor exhibiting the most improvement at each iteration and uses a stop criterion of local optimality. We used the BRKGA to compute initial solutions, and the local search found no improvements compared to our BRKGA for all instances; thus, all the solutions obtained by our BRKGA are local optima. Even when local optimality is a strong characteristic of solution quality, we compare our BRKGA to a Genetic Algorithm for larger instances in the next section.

6.4 Solving large instances using the BRKGA and a Genetic Algorithm

The above numerical results show that the CPLEX commercial solver can obtain feasible solutions for instances of types A, B, and C, even when there is a limitation imposed in terms of computational times. However, the solver cannot obtain feasible solutions for large instances (of 400 locations and 1200 demand zones). We define these instances as type D, and details of the parameters involved are as follows: $|I| = 400$, $p = 50$, $|J| = 1200$, $s_i = 2$, $\varepsilon_i = 5$, $m_j = 15$, and $l = 75$. We perform a comparison of our BRKGA and a Genetic Algorithm to test the efficiency of BRKGA with regard to larger instances. The Genetic Algorithm uses the same solution encoding and crossover as the BRKGA, but we use a probability of 0.5 of inheriting an element from one parent. The mutation randomly chooses 5% of the elements in the child random-key of the crossover and randomly modifies those elements. The elitism in the GA is to preserve 10% of the best elements in the next population. We also implement

a population size of 500 individuals and a stop criterion of 100 iterations with no improvement. Figure 8 presents the value of the accessibility measure for the 30 instances of type D.

Notice that the BRKGA outperforms the GA in solution quality for all the instances. The numerical results exhibit a relative improvement in the solution quality of 12.93%, for the BRKGA over the GA. Moreover, the average computational time of the BRKGA is 23.74 min, as compared to 35.34 min of execution time for the GA. Hence, our BRKGA is more efficient than the proposed GA for large instances.

To summarize, as we have seen previously, our BRKGA is capable of obtaining solutions with a quality comparable with those of a commercial solver's solutions but using only a fraction of computational time. Moreover, the solutions found by our BRKGA are, in general, local optima with respect to $2\text{-}opt$ neighborhoods. And finally, the BRKGA outperforms a classic Genetic Algorithm for large instances. Indeed, those large instances are intractable by CPLEX since the solver can not find even a feasible solution due to hardware limitations. Thus, we have developed an optimization approach that can be used for the decision-making process during a pandemic for large instances. The following section shows an example of a potential implementation of our methodology in a real scenario in Mexico.

6.5 Implementation on a scenario assuming reduced mobility due to the COVID-19 pandemic

Sections 6.2 and 6.3 provide an analysis of our BRKGA in terms of solution quality and computational time, as compared with the CPLEX commercial solver and a Genetic Algorithm. This section analyzes a potential implementation of our optimization approach for a real-world scenario.
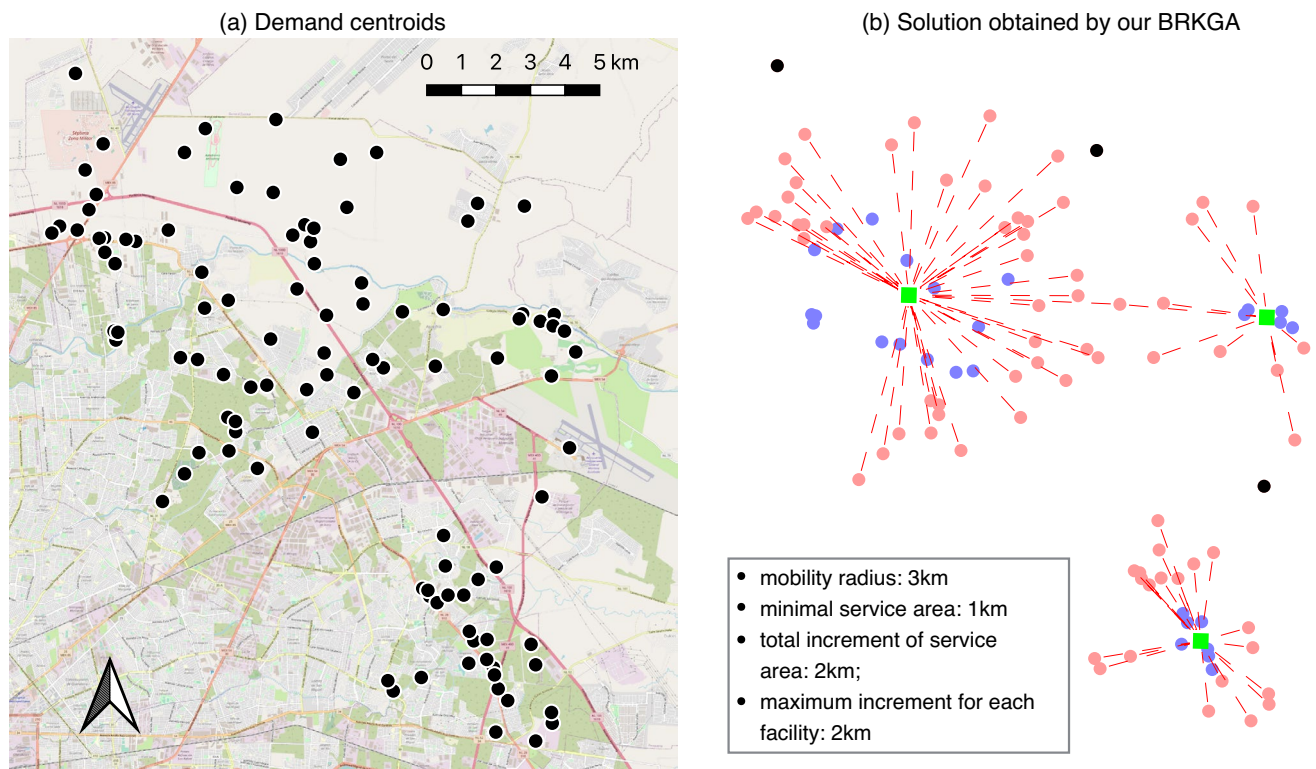
Fig. 9 Demand centroids and a solution for our case study. Data extracted from [36] and background map obtained from ©OpenStreetMap contributors

In particular, we set a mobility radius for inhabitants in demand centroids J, and we calculated the number of mobile labs needed. The population within J has access to COVID-19 testing, assuming reduced mobility due to pandemic-related restrictions. There are 51 municipalities in the state of Nuevo Leon. We focus on marginal localities of the municipality of Apodaca, which was third in the ranking of higher infection rates during March-June of 2020. In particular, there are 104 demand centroids, which are also represented by the location set I. The rest of the parameters are set as follows: a minimum service area of 1 km for mobile labs; a maximum increase in the service area of 2km for each lab; finally, a mobility radius of 3km.

To find the number of mobile labs, we vary the parameter, p, from 5 to 1. The total maximum increase in the lab's service radius is set as $l = p - 1$. Figure 9a shows the localities in the municipality of Apodaca that represent the demand centroids. Figure 9b shows the feasible solution obtained by our BRKGA for a parameter of $p = 3$: the squares represent the mobile labs; the blue circles represent the covered demand centroids; the red circles are the uncovered demand centroids with access to the service; the black circles denote the demand centroids without access to COVID-19 testing; and finally, the dashed lines point out the opportunities.

Notice that only three demand centroids have no access to the service, i.e., assuming a mobility radius of three km for potential clients, three mobile laboratories with a service area of one kilometer are sufficient to guarantee access to COVID-19 tests for 97.11% of the demand centroids. Figure 10 shows two alternative solutions using a service radius of $s = 2$ kilometers (case (a)), and using one more mobile lab $p = 4$ (case (b)). We highlight that assuming a minimal service area of 2 km (case (a)) leads to access for 100% of the inhabitants at the demand centroids while increasing the number of mobile labs does not achieve the same outcome.

In general, for a predefined level of client mobility, our optimization approach can be used as a diagnostic tool to estimate the number of mobile labs and determine the locations where they should be based in order to optimize access to a COVID-19 testing service. Since our solution algorithm takes only seconds of computing time, this analysis can be performed for other municipalities and for a situation whereby the laboratories' locations are changed throughout the days (taking advantage of the labs' mobility). Finally, our approach could be enhanced by taking into account information from areas with high rates of infections and preferences of the decision-maker to prioritize some accessibility indicators depending on the context of the practical implementation.
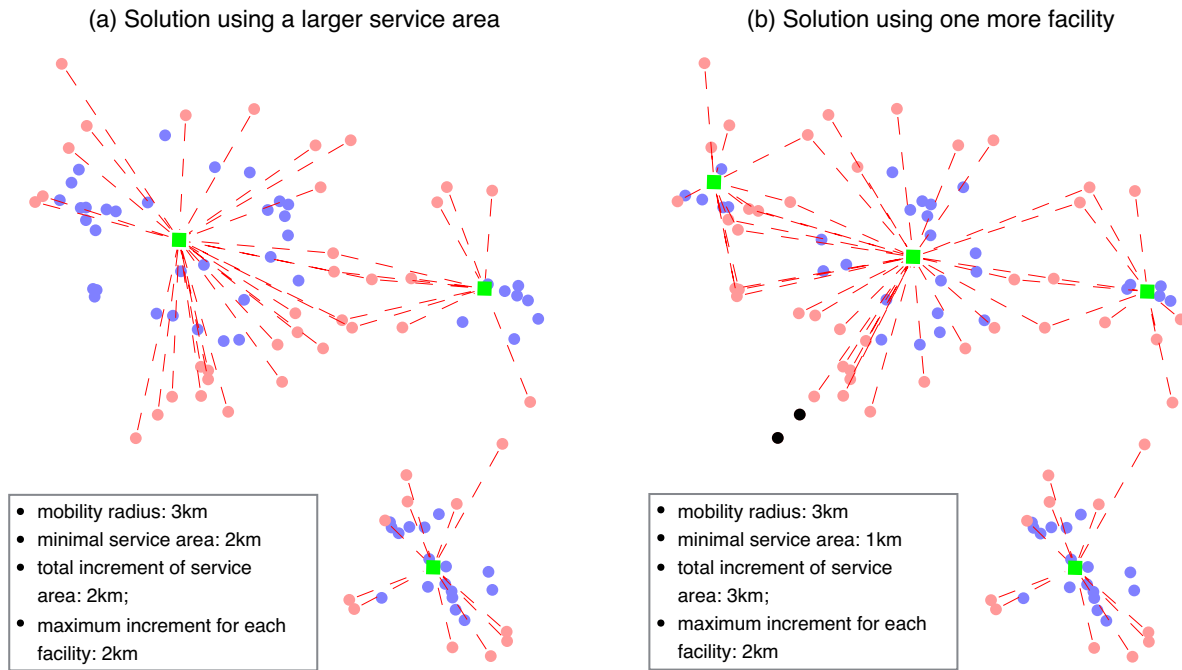
(a) Solution using a larger service area

- mobility radius: 3km
- minimal service area: 2km
- total increment of service area: 2km;
- maximum increment for each facility: 2km

(b) Solution using one more facility

- mobility radius: 3km
- minimal service area: 1km
- total increment of service area: 3km;
- maximum increment for each facility: 2km

Fig. 10 Solutions using a larger service area and 4 mobile laboratories

7 Conclusions

The COVID-19 outbreak led to eventualities such as the restricting of the population's mobility to foster social distancing and the planning of the logistics for COVID-19 testing. These are appropriate circumstances for the requirement to implement a practical way to solve the mobile labs location problem, optimizing an accessibility measure since accessibility focuses on guaranteeing more and better opportunities whereby potential clients can reach a service without traveling long distances. In response, we introduced a variant of the location problem, which optimizes accessibility indicators based on classical, gradual, and cooperative coverage and considers the variable service areas of mobile labs for COVID-19 testing. Our main goal was to support the necessary decision-making process by defining an efficient optimization approach in terms of solution quality and computational time. Even when we formulate our optimization problem using a mixed-integer linear program, commercial solvers are not capable of obtaining high-quality solutions for large instances in reasonable computational times, and the decision-making can arise several times during a pandemic since we focus on mobile labs, which can be re-located considering the infection rate in specific regions of a city. Thus, more efficient algorithms are needed, so we designed a Biased Random-Key Genetic Algorithm (BRKGA) to fulfill this requirement.

Numerical results relating to randomly generated instances show that our proposed BRKGA obtains solutions with a similar quality to those found by the commercial solver of CPLEX 2.9 when it is limited to one hour of computational time, but BRKGA uses only a fraction of this execution time. Moreover, our BRKGA outperforms CPLEX under the assumption of the availability of only short computational times; also, it obtains better solutions than a Genetic Algorithm when solving larger instances that are intractable in relation to the commercial solver (no feasible solution reported by the solver of CPLEX due to hardware limitations). Therefore it can be seen that our optimization approach is a tool which can be implemented in a real context. The latter is illustrated by some experimental work based on the allocation of mobile laboratories for COVID-19 testing in Nuevo Leon, Mexico. The results show that our optimization approach is helpful as a diagnostic tool to determine the number of mobile labs which can guarantee accessible COVID-19 testing, while the hard constraint of reduced mobility for the purposes of social distancing is in place.

Further research areas would be: to determine a generalization of the problem, which considers: non-unitary demands, facility capacities, and other elements such as alternative definitions of client mobility in terms of space and time.

Table 7 Numerical results of our BRKGA using different combinations of parameters

Combinations of parameters				Instances A		Instances B		Instances C	
pop	*pe*	*pm*	*ph*	Obj	CPUtime	Obj	CPUtime	Obj	CPUtime
100	0.15	0.05	0.50	0.4143	1.3532	0.4490	5.6924	0.4656	51.4178
100	0.20	0.10	0.60	0.4154	1.2014	0.4532	6.331	0.4650	98.7232
200	0.15	0.05	0.50	0.4158	2.0424	0.4559	10.1174	0.4714	81.105
300	0.25	0.22	0.70	0.4162	3.1468	0.4589	25.832	0.4747	192.276
300	0.25	0.20	0.70	0.4167	2.8312	0.4613	16.4022	0.4781	153.4106
300	0.15	0.05	0.50	0.4186	2.8564	0.4620	17.0858	0.4799	137.1756
300	0.25	0.15	0.70	0.4196	6.7418	0.4609	31.9614	0.4788	282.5774
300	0.20	0.10	0.60	0.4181	6.4334	0.4544	38.1148	0.4788	326.8602
300	0.20	0.10	0.70	0.4185	12.3622	0.4639	59.1856	0.4837	252.1418
300	0.20	0.10	0.50	0.4191	3.5512	0.4619	18.4534	0.4813	172.2688
300	0.25	0.10	0.50	0.4191	3.6738	0.4618	14.8830	0.4783	119.6560
300	0.25	0.15	0.50	0.4179	3.6338	0.4599	21.0138	0.4808	183.2870
300	0.25	0.05	0.50	0.4195	3.2818	0.4594	14.3252	0.4783	136.5150
300	0.20	0.05	0.50	0.4190	3.4226	0.4605	14.1120	0.4796	146.8208
400	0.15	0.05	0.50	0.4186	9.2048	0.4587	62.0236	0.4816	399.6076
500	0.15	0.05	0.50	0.4195	4.8358	0.4650	28.2388	0.4816	153.0952
500	0.25	0.15	0.70	0.4195	8.6982	0.4630	70.4054	0.4827	215.9218
500	**0.20**	**0.10**	**0.60**	**0.4189**	11.8910	**0.4634**	60.3002	**0.4833**	488.4736
500	0.20	0.10	0.70	0.4185	6.0840	0.4586	44.3084	0.4732	274.5906
I + *p*	0.15	0.05	0.50	0.4114	0.9326	0.4494	7.0230	0.4714	100.071
I + *p*	0.25	0.20	0.70	0.4132	0.7516	0.4512	5.0686	0.4744	92.2416
2(*I* + *p*)	0.15	0.05	0.50	0.4104	1.0266	0.4549	11.6370	0.4819	207.6846
2(*I* + *p*)	0.25	0.20	0.70	0.4136	1.3216	0.4551	10.9768	0.4797	184.7594
3(*I* + *p*)	0.15	0.05	0.50	0.4128	1.3930	0.4580	11.1308	0.4841	242.8594
3(*I* + *p*)	0.25	0.20	0.70	0.4137	1.6116	0.4562	19.7482	0.4862	326.3666

Appendix A: Parameter calibration for the BRKGA

Like many other metaheuristic algorithms, our BRKGA is defined in terms of the setting of a number of parameters, and different values of those parameters affect the behavior of our algorithm in different ways. In response, we used multiple combinations of these parameter values in our testing, including setting the instances' sizes. Specifically, we solved five instances of types A, B, and C for all the combinations of parameters. Table 7 shows the average of the accessibility measure (denoted as "Obj") and the average computational time in seconds for each instance type (column "CPUtime"). We highlight the best and worst values with blue and red fonts, respectively.

Notice that there is no clear dominance for any combination of parameters. Hence, we chose the following parameter values (indicated by bold font) since they lead to almost the best solution quality for instances types A, B, and C: i.e., a population size of $pop = 500$; a fraction of elite parents in the population of $pe = 0.20$; a fraction

of individuals in the population replaced by mutations of $pm = 0.10$; a probability of inheriting characteristics of the elite parent during crossover of $ph = 0.60$; and a stop criterion of 100 iterations with no improvement. The number of iterations in the stop criterion is not shown in Table 7 since we obtain similar results.

Appendix B: Solutions obtained by CPLEX for instances based on the case study in Nuevo Leon, Mexico

The comparison of CPLEX and our BRKGA was presented in Sections 6.2 and 6.3, while Section 6.5 is focus on analyzing the potential implementation of our proposed metaheuristic algorithm on a real-world scenario. However, we now present the solutions obtained for the three instances based on the case study but using the solver of CPLEX.

We highlight that computational times to obtain solutions in Figs. 11, 12, and 13, were of 138s, 12.45s, and 13.51s,
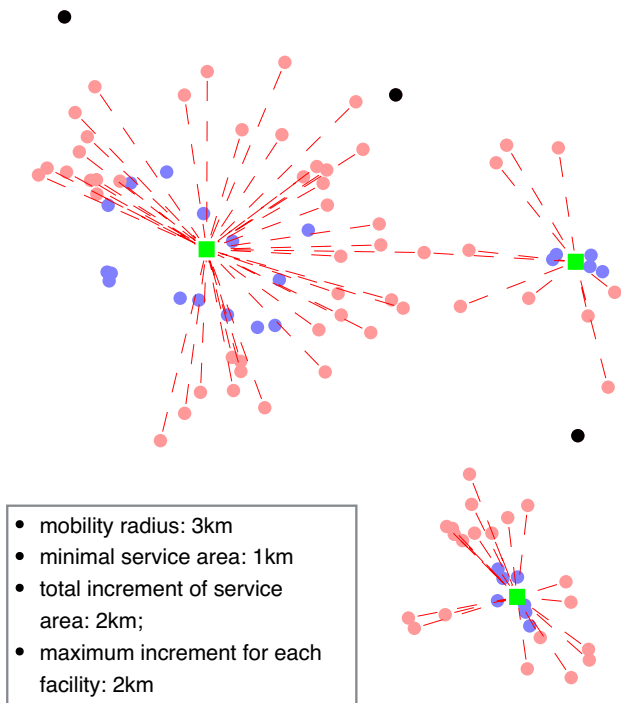
- mobility radius: 3km
- minimal service area: 1km
- total increment of service area: 2km;
- maximum increment for each facility: 2km

Fig. 11 Solution obtained with CPLEX for the first instance based on the case study

- mobility radius: 3km
- minimal service area: 1km
- total increment of service area: 3km;
- maximum increment for each facility: 2km
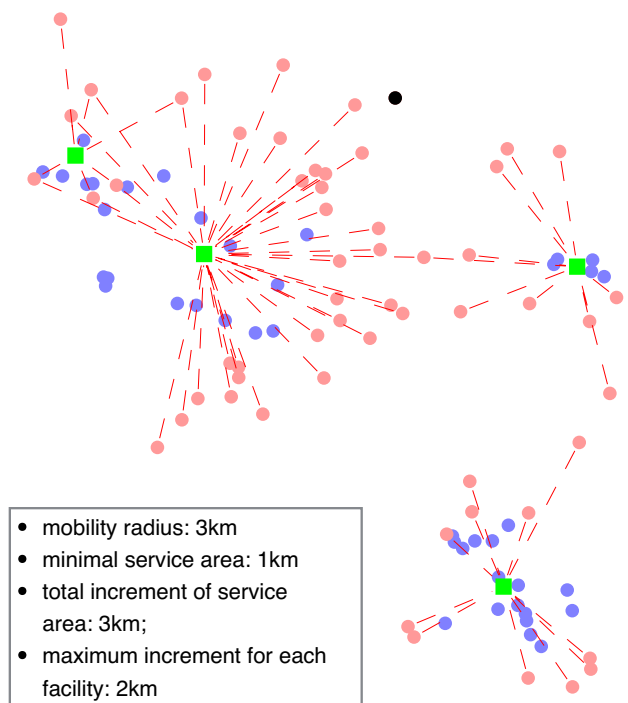
Fig. 13 Solution obtained with CPLEX for the third instance based on the case study

- mobility radius: 3km
- minimal service area: 2km
- total increment of service area: 2km;
- maximum increment for each facility: 2km
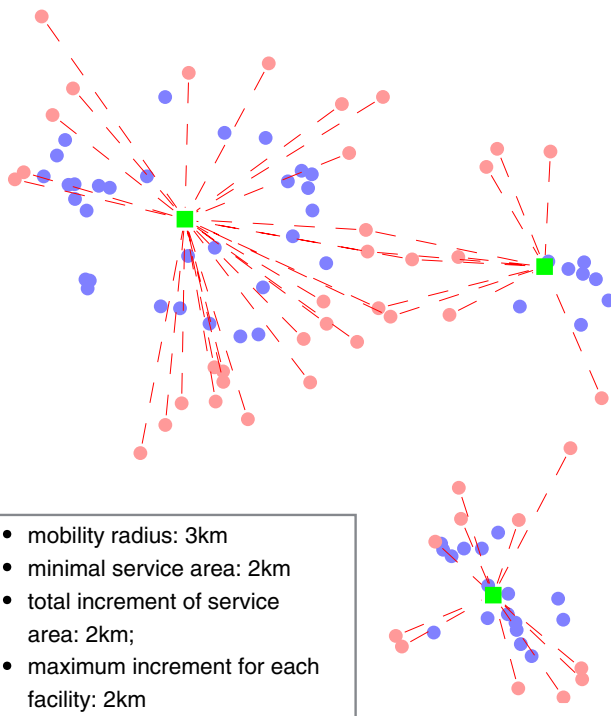
Fig. 12 Solution obtained with CPLEX for the second instance based on the case study

respectively. Moreover, the only difference between the solutions obtained with CPLEX and the ones generated by our BRKGA is that we obtain a solution with one more demand point with access to the service when solving the instance of four facilities with the solver of CPLEX (see Fig. 13).

Declarations

Conflict of Interests The authors declare no conflict of interest of any kind.

References

1. Balasa A (2020) COVID-19 on lockdown, social distancing and flattening the curve - a review. Eur J Bus Manag Res 5(3):1–4
2. Lewnard J, Nathan C (2020) Scientific and ethical basis for social-distancing interventions against COVID-19. The Lancet Infect Dis 20(6):631–633
3. European Centre for disease prevention and control (2020) Testing strategies. https://www.ecdc.europa.eu/en/covid-19/surveillance/testing-strategies
4. EL UNIVERSAL (2020) Nuevo León alcanzará a Corea del Sur en número de pruebas de COVID-19. https://www.eluni

versal.com.mx/estados/nuevo-leon-alcanzara-corea-del-sur-en-numero-de-pruebas-de-covid-19

5. Guadalupe Gobierno Municipal (2020) Realizan en Guadalupe pruebas de COVID-19 en laboratorio móvil. https://www.guadalupe.gob.mx/realizan-en-guadalupe-pruebas-de-covid-19-en-laboratorio-movil/. https://www.eluniversal.com.mx/estados/nuevo-leon-alcanzara-corea-del-sur-en-numero-de-pruebas-de-covid-19

6. Jasso-Díaz D (2021) Un problema de localización y asignación aplicado al sistema de salud durante la pandemia por el COVID-19. Master's thesis, Universidad Autónoma de Nuevo León

7. Geurs K, van Eck J (2001) Accessibility measures: Review and applications. Evaluation of accessibility impacts of land-use transport scenarios, and related social and economic impacts. Tech. Rep RIVM Report 408505 006,, National Institute of Public Health and the Environment, Utrecht University, Bilthoven/Utrecht, Netherlands

8. Makrí M, Folkesson C (1999) Accessibility measures for analyses of land use and travelling with geographical information systems. In: Proceedings of the 2nd KFB-research conference, Lund, Sweden, pp 1–17

9. Ibarra-Rojas O, Hernandez-Landa L, Ozuna L (2018) The accessibility vehicle routing problem. J Clean Prod 172:1514–1528

10. Ibarra-Rojas O, Ozuna L, López-Piñón D (2019) The maximal covering location problem with accessibility indicators. Socio-Economic Planning Sciences, p 100758

11. Gu W, Wang X, McGregor S (2010) Optimization of preventive health care facility locations. Int J Health Geogr 9:1–17

12. Stummer C, Doerner K, Focke A, Heidenberger K (2004) Determining location and size of medical departments in a hospital network: A multiobjective decision support approach. Health Care Manag Sci 7(1):63–71

13. Yin X, Buyuktahtakin I (2021) A multi-stage stochastic programming approach to epidemic resource allocation with equity considerations. Health Care Manag Sci 24(3):597–622

14. Ahmadi-Javid A, Seyedi P, Syam SS (2017) A survey of healthcare facility location. Comput Oper Res 79:223–263

15. Farahani R, Asgari N, Heidari N, Hosseininia M, Goh M (2012) Covering problems in facility location: A review. Comput Ind Eng 62:368–407

16. Doerner K, Focke A, Gutjahr WJ (2007) Multicriteria tour planning for mobile healthcare facilities in a developing country. Eur J Oper Res 179(3):1078–1096

17. Boonmee C, Arimura M, Asada T (2017) Facility location optimization model for emergency humanitarian logistics. Int J Disaster Risk Reduction 24:485–498

18. Abounacer R, Rekik M, Renaud J (2014) An exact solution approach for multi-objective location–transportation problem for disaster response. Comput Oper Res 41:83–93

19. Barzinpour F, Esmaeili V (2014) A multi-objective relief chain location distribution model for urban disaster management. Int J Adv Manuf Technol 70:1291–1302

20. Jia H, Ordóñez F, Dessouky M (2007) Solution approaches for facility location of medical supplies for large-scale emergencies. Comput Ind Eng (2), 257–276

21. Murali P, Ordóñez F, Dessouky MM (2012) Facility location under demand uncertainty: Response to a large-scale bio-terror attack. Socioecon Plann Sci 46(1):78–87

22. Salman F, Yücel E (2015) Emergency facility location under random network damage: insights from the Istanbul case. Comput Oper Res 62:266–281

23. Gendreau M, Laporte G, Semet F (2001) A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. Parallel Comput 27(12):1641–1653

24. Doerner K, Gutjahr W, Hartl R, Karall M, Reimann M (2005) Heuristic solution of an extended double-coverage ambulance location problem for Austria. Central Eur J Oper Res 13(4):325–240

25. Erdemir ET, Batta R, Rogerson PA, Blatt A, Flanigan M (2010) Joint ground and air emergency medical services coverage models: A greedy heuristic solution approach. Eur J Oper Res 207(2):736–749

26. Grot M, Becker T, Steenweg PM, Werners B (2021) Enhanced coverage by integrating site interdependencies in capacitated EMS location models. Health Care Management Science. Advance online publication. https://doi.org/10.1007/s10729-021-09562-4

27. Mousavi H, Darestani SA, Azimi P (2021) An artificial neural network based mathematical model for a stochastic health care facility location problem. Health Care Manag Sci 24(3):499–514

28. Drezner Z, Wesolowsky G, Drezner T (2004) The gradual covering problem. Nav Res Logist 51(6):841–855

29. Azizan M, Go T, Hatta W, Lim C, Teoh S (2017) Comparison of emergency medical services delivery performance using maximal covering location and gradual cover location problems. Int J Electr Comput Eng 7(5):2791–2797

30. Jabalameli M, Behzad B, Mohammad M (2010) Capacitated facility location problem with variable coverage radius in distribution system, pp 231–237

31. Eydi A, Torabi H (2019) A maximal covering problem in supply chain considering variable radius and gradual coverage with the choice of transportation mode. Arab J Sci Eng 44(8):7219–7233

32. Bagherinejad J, Bashiri M, Nikzad H (2018) General form of a cooperative gradual maximal covering location problem. J Ind Eng Int 14(2):241–253

33. Karatas M (2017) A multi-objective facility location problem in the presence of variable gradual coverage performance and cooperative cover. Eur J Oper Res 262(3):1040–1051

34. Gonçalves J, Resende M (2011) Biased random-key genetic algorithms for combinatorial optimization. J Heuristics 17(5):487–525

35. Resende M (2012) Biased random-key genetic algorithms with applications in telecommunications. TOP 20(1):130–153

36. INEGI (2020) Catálogo único de claves de áreas geoestadísticas estatales municipales y localidades. https://www.inegi.org.mx/app/ageeml/