

The maximal covering location problem with accessibility indicators and mobile units



Salvador J. Vicencio-Medina^a, Yasmin A. Rios-Solis^{a,*}, Omar Jorge Ibarra-Rojas^b,
Nestor M. Cid-Garcia^c, Leonardo Rios-Solis^{d,e,f}

^a School of Engineering and Sciences, Tecnológico de Monterrey, Ave. Eugenio Garza Sada 2501, Monterrey, 64849, N.L, Mexico

^b Universidad Autónoma de Nuevo León, Facultad de Ciencias Físico Matemáticas, San Nicolás de los Garza, 66455, Nuevo León, Mexico

^c Laboratorio Nacional de Geointeligencia, CONACYT-Centro de Investigación en Ciencias de Información Geoespacial, Aguascalientes, Aguascalientes, Mexico

^d Institute for Bioengineering, School of Engineering, University of Edinburgh, Edinburgh, EH9 3BF, UK

^e Centre for Synthetic and Systems Biology (SynthSys), University of Edinburgh, Edinburgh, EH9 3BF, UK

^f School of Natural and Environmental Sciences, Molecular Biology and Biotechnology, Newcastle University, Newcastle Upon Tyne, NE1 7RU, UK

ARTICLE INFO

Dataset link: <https://doi.org/10.6084/m9.figshare.20460867.v3>

Keywords:

Facility location problem

Matheuristic

COVID-19

Mixed-integer linear program

Accessibility

Estimation of distribution algorithm

ABSTRACT

We study the Maximal Covering Location Problem with Accessibility Indicators and Mobile Units that maximizes the facilities coverage, the accessibility of the zones to the open facilities, and the spatial disaggregation. The main characteristic of our problem is that mobile units can be deployed from open facilities to extend the coverage, accessibility, and opportunities for the inhabitants of the different demand zones. We formulate the Maximal Covering Location Problem with Accessibility Indicators and Mobile Units as a mixed-integer linear programming model. To solve larger instances, we propose a *matheuristic* (combination of exact and heuristic methods) composed of an Estimation of Distribution Algorithm and a parameterized Maximal Covering Location Problem with Accessibility Indicators and Mobile Units integer model. To test our methodology, we consider the Maximal Covering Location Problem with Accessibility Indicators and Mobile Units model to cover the low-income zones with Severe Acute Respiratory Syndrome Coronavirus 2 patients. Using official databases, we made a set of instances where we considered the poverty index, number of population, locations of hospitals, and Severe Acute Respiratory Syndrome Coronavirus 2 patients. The experimental results show the efficiency of our methodologies. Compared to the case without mobile units, we drastically improve the coverage and accessibility for the inhabitants of the demand zones.

1. Introduction

The Maximal Covering Location Problem determines the location of facilities and the client-facility assignments, considering several assumptions, e.g., fixed costs for locating the facilities and the transportation costs from facilities to clients [1–3]. Indeed, the MCLP has been widely studied in the context of the supply chain [4], but also in scenarios of humanitarian logistics [5] since it can potentially focus on increasing a social benefit measure (instead of costs minimization). Moreover, several variants of the MCLP arise to define a representative decision problem in different scenarios, such as the location of emergency vehicles [6], security stations [7], and health-care centers [8], among others.

The Maximal Covering Location Problem with Accessibility indicators (MCLPA), proposed in Ibarra-Rojas et al. [9] is a variant of the MCLP, which optimizes a generalized coverage function in terms of several accessibility indicators and assumes a service radius of facilities

and a mobility area for inhabitants of demand zones. The MCLPA aims to provide several well-located opportunities for people to get a service or satisfy their demands. In particular, the authors propose optimizing the following six accessibility indicators. (i) *Coverage*: determines the number of demand zones covered by the service area of the facilities, (ii) *Minimum access*: determines the total of zones covered by the service network, either with the service area of facilities or considering the mobility area of inhabitants, (iii) *Mobility costs*: represent the mobility costs for the inhabitants of not covered zones to seek a facility within their mobility area, (iv) *The proximity of the service*: represents the distance from not covered zones to their closest facility. If the service area of the facilities or the mobility area of the zones is extended, then the clients can be satisfied, (v) *Number of opportunities*: represents the number of locations within the mobility area of inhabitants of a demand zone, and (vi) *Geographical segregation*: determines the degree

* Corresponding author.

E-mail address: yasmin.riosolis@tec.mx (Y.A. Rios-Solis).

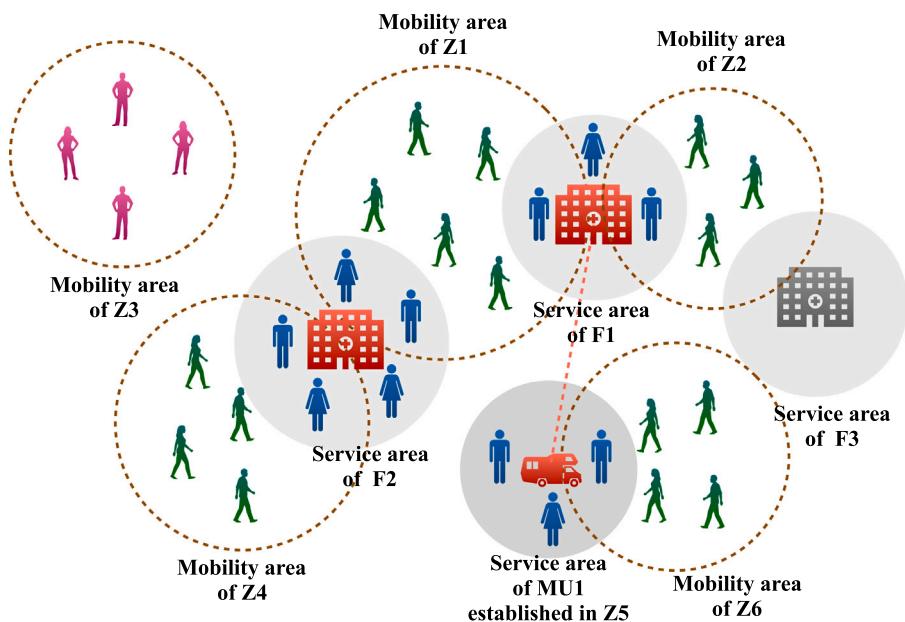


Fig. 1. General representation for the MCLPAMU with mobile units.

of dispersion between the zones without service or the possibility of being satisfied.

In this article, we present an extension for the MCLPA, adding mobile units (MU) to increase the service area of facilities as well as to optimize the accessibility indicators. We call this problem the Maximal Covering Location Problem with Accessibility indicators and Mobile Units (MCLPAMU). A MU is a transport unit (bus, car, camper) that extends the number of opportunities from the open facilities to the clients. Each facility or mobile unit can determine its service area and the maximum population assigned (demand), where the demand or service area can differ for each facility and/or mobile unit. Furthermore, each customer could or could not have access to some facility or mobile unit considering their mobility area. This is determined by considering the maximum travel time or distance from the clients to some facility or mobile unit. Generally, the clients have different mobility areas; therefore, some can access more than one facility or mobile unit. The MCLPAMU considers a set of clients as a demand zone, and it sets the centroid of this zone as the same distance for each customer to some facility. It assumes a mobility area for inhabitants of each demand zone. The difficulty in solving the MCLPAMU arises from the MU, but the results show the large impact of considering them in an integral methodology.

Fig. 1 shows a general representation of the MCLPAMU considering three possible locations to open a facility (F1-F3) with their corresponding service area (gray circles) and six demand zones (Z1-Z6) with their respective mobility area (brown dashed circle). The service area determines the covered population for each facility or mobile unit (client symbol in blue color). Notice the first two locations are selected to open a facility (F1 and F2), and a mobile unit (MU1) that belongs to F1 is established in Z5, which means that MU1 already covers the zone and could therefore extend the service to other zones (like Z6). The customers in Z1, Z2, Z4, and Z6 (client symbol in green color) can reach the service because they have access to some facility or mobile unit (the mobility area of Z1 has access to both facilities). However, the customers in Z3 (client symbol in pink color) cannot reach the service of the facilities or mobile units, even considering its mobility area.

The FLP has several real-life applications for the medical, industrial, academic, and logistic sectors. The MCLPAMU can extend all these applications by maximizing the demand to satisfy by using mobile units. In this study, we present a real-life implementation for the MCLPAMU to support decision-making in the medical sector, which attends to the

pandemic effects caused by the SARS-CoV-2 (Severe Acute Respiratory Syndrome Coronavirus 2), better known as COVID-19 virus. COVID-19 was detected in Wuhan, China, and has become a global challenge for many countries' healthcare systems, causing 6.5 million deaths, and 585 million more have tested positive for the virus. The countries of the USA, India, Brazil, France, and Germany have been most affected by some variants of COVID-19 in terms of the total number of deaths recorded [10]. Implementing comprehensive testing and an adequate track and trace process for positive patients is essential to stop the spread of the disease, leading to high demand for diagnostic testing with good accessibility for the population [11]. While reducing the cost of SARS-CoV-2 diagnostics has contributed significantly to meeting this demand and increasing the testing capacity [12], these diagnostic resources must be optimally managed in terms of location for test centers and availability of testing capacity, among other factors. This can be of paramount importance, especially in low-income communities, to establish an effective pandemic control strategy.

The implementation of the MCLPAMU for the medical services of Mexico uses COVID-centers (hospitals or mobile units) to collect COVID-19 tests to detect the virus propagation faster. An opportune reaction of the medical services plays a vital role in the attention of the pandemic effects because it minimizes the virus propagation and avoids significant damage to the community. To reach this goal, the MCLPAMU considers several key elements that must be adapted to the problem, e.g., choosing a determined number of hospitals established as COVID-19 centers (facilities), specifying the service area for each hospital, the maximum number of people that could be satisfied, and the number of mobile units (ambulances) for each one; as well as determining the demand zones (municipalities, colonies, etc.) including their mobility area.

2. Related literature

Since we addressed a variant of the MCLP to optimize an accessibility measure for medical services, our literature review focused on the following three aspects: the relevance of the Maximal Covering Location Problem (MCLP) in scenarios aiming to optimize a social impact, the different variants of the MCLP due to different characteristics in the case study, and finally, the implementation of the MCLP in the context of the COVID-19 pandemic.

Due to the nature of the MCLP of maximizing a coverage measure of people instead of cost minimization, several studies using the MCLP aimed to optimize a measure of social benefit, mainly in the context of humanitarian logistics (HL). In HL, the main goal is to provide a service or fulfill the demand of people affected by natural or human-caused disasters [see the review of 13, about facility location problems in HL]. Location problems arise from the need to locate emergency vehicles or mobile units [see 14, 15] and healthcare centers such as hospitals and temporary medical services [see a review of the Healthcare Facility Location Problem in 16]. Studies like Stummer et al. [17] and Doerner et al. [18] address maximal coverage location problems for healthcare centers using a classic definition of coverage, i.e., a demand point is covered when it is within a predefined maximum distance (or service radius) from facilities. Moreover, the authors also considered other objectives, such as distance to the service and operational costs. However, we highlight that different measures of coverage for the MCLP should be implemented in the context of humanitarian relief.

Indeed, Ahmadi-Javid et al. [16] stated that “relevant characteristics should be considered in location modeling to make location models more efficient”. For example, the concept of multi-coverage represents the case when a demand point is covered when it is within the service radius of more than one facility, which can be relevant when locating mobile units for healthcare services [e.g., 19–21] or security stations [e.g., 22]. Recently, the study of Grot et al. [15] addressed a location problem for emergency health centers to maximize the covered demand. The authors assumed that an emergency center might be busy receiving an aid call (due to its limited capacity). So the service may be forwarded to another center, defined as inter-dependency. Then, it must be considered that clients are covered if they can be satisfied within a specific limited time response and considering those interdependencies instead of a classic 0–1 coverage. The authors implemented a commercial solver on a limited scenario to show the effectiveness of their proposed approach in terms of a faster response. Another example is the optimization approach of Mousavi et al. [14], which focused on the location of trauma centers and ambulance stations (ambulances are also assigned to trauma centers) to minimize operation costs of satisfying all demand points. The authors assumed that patients could be transported to trauma centers by ambulance (direct service of facilities) or helicopter; thus, a demand point was covered when inhabitants of that point could access a trauma service. Studies similar to the last one are presented in [23] where the authors want to optimize three objective functions: (i) minimize the sum of all transportation durations, (ii) minimize the number of people needed to operate a health center, and (iii) minimize the total uncovered demand. In this case, the author assumes that every demand point is accessible. It means it can be covered by at least one health center with a time window. Another similar study with a military application is presented in Jenkins et al. [24]. In this study, the authors want to maximize the expected demand, minimize the number of located mobile aeromedical facilities in any deployment phase, and minimize the total of mobile aeromedical facilities relocations throughout the deployment. The authors assumed that each aeromedical facility has sufficient resources and capacity to handle all incoming demand.

On the other hand, the classic 0–1 coverage function may be too restrictive since the level of coverage may depend on the distance from a demand point to the service network; thus, a “gradual coverage” function is needed. For example, Azizan et al. [25] studied the location of ambulance centers to maximize demand satisfaction in terms of two service radii, i.e., there is complete demand satisfaction if a demand point is within a predefined service radius and the level of coverage decreases to 0 when the distance reaches a maximal distance. The authors implemented a particle swarm optimization algorithm to solve real-size instances based on Malaysia’s Johor Bahru region data.

Since there is no unique definition of coverage to guarantee the accessibility of the service network, several generalizations of coverage measures can be proposed (e.g. [26]). For example, Forghani Youshanlo

and Sahraeian [27] addressed a multi-objective location problem to maximize single and double coverage in terms of a gradual coverage function versus minimizing the operational costs and not-covered demand. A lexicographic multi-objective linear programming approach was used to represent the decision-makers subjective preferences and propose a single feasible solution. Karatasa et al. [28] also defined a location problem to optimize a weighted objective function based on coverage, operational costs, and work balance among open facilities. The proposed problem integrated gradual and cooperative coverage and determined the service radii of facilities to extend the coverage of the service network. The authors implemented a two-stage approach that obtains high-quality solutions for large instances in acceptable computational times. Finally, Ibarra-Rojas et al. [9] proposed a location problem to maximize a generalized coverage function in terms of accessibility indicators such as classic coverage, travel cost in terms of multi/gradual coverage, number of opportunities, distance to the service, and geographical segregation. The authors implemented a commercial solver to obtain optimal solutions for large instances. Even when the mentioned studies did not focus on the location of healthcare facilities, the literature review of Boonmee et al. [13] addressed the need to improve accessibility to humanitarian aid concerning classical coverage. In response to this, in this work, we generalized the approach of Ibarra-Rojas et al. [9] by considering the location of mobile units to support the decision-making process of locating medical services in the context of the COVID-19 pandemic.

Recently, location problems have been implemented to foster medical services, testing procedures, and vaccination centers during the COVID-19 pandemic [e.g., 29, in Nigeria]. Moreover, the accessibility of such services is essential due to restrictive conditions such as social distancing and low mobility. For example, Hassan et al. [30] implemented a classic MCLP for Field Hospitals to maximize demand coverage subject to a maximal number of hospitals. Calderon et al. [31] focused on the location of ambulance stations to optimize a weighted sum of demand coverage, where the weights of demand points were defined by combining a socio-economic indicator and a disease rate. The authors assumed a heterogeneous fleet of ambulances assigned to the upper bounds of the stations, ambulances assigned to each station, and ambulances assigned to the entire system. The solver of GUROBI was then used to find optimal solutions for instances based on real data from the metropolitan region of Chile. Another example was the two-stage programming approach of Kuvvetli [32] for a location-allocation problem that determines the location of capacitated testing facilities to detect the COVID-19 disease and the client-facility-laboratory assignments. The following objectives were considered: (i) maximum accessibility aimed to locate the test sampling facilities centers at minimum distances to the neighborhoods; (ii) number of testing facilities; and (iii) distance from testing facilities to laboratories. The author implemented the GAMS modeling language and the SCIP optimization solver to find optimal solutions for a real scenario in Istanbul and Adana, Turkey using less than six hours of computational time. Finally, [33] defined a bi-objective MCLP to determine p capacitated vaccination centers maximizing the demand coverage versus minimizing the traveled distance to access the service. The authors assumed that multiple vaccination centers (partial coverage) could satisfy the demand points. They implemented the solver of GUROBI in a lexicographic approach to find optimal solutions for instances based on data from Yogyakarta, Indonesia.

Table 1 summarizes the studies in the literature considering different coverage concepts such as classic, cooperative, gradual, or partial coverage. Column 6 indicates the implementation or not of mobile units. Column 7 shows the type of objective function: with a single objective (SO) or multi-objective (MO).

In summary, we propose a novel variant of the MCLP to determine healthcare centers and mobile units that optimize a generalized coverage function in terms of different accessibility indicators. Those indicators are relevant from a social perspective. We implemented our approach in the context of the COVID-19 pandemic to provide good and different opportunities for people to get a COVID-19 testing service.

Table 1
Summarize of literature review.

Study	Classic coverage	Cooperative coverage	Gradual coverage	Partial coverage	Mobile units	Objective
					SO	MO
Stummer et al. [17]	✓					✓
Doerner et al. [18]	✓					✓
Gendreau et al. [19]		✓				✓
Doerner et al. [20]		✓				✓
Erdemir et al. [21]		✓				✓
Grot et al. [15]		✓				✓
Mousavi et al. [14]		✓				✓
Abounacer et al. [23]	✓				✓	✓
Jenkins et al. [24]			✓		✓	✓
Azizan et al. [25]			✓			✓
Forghani Youshanlo and Sahraeian [27]			✓			✓
Karataş et al. [28]		✓	✓			✓
Ibarra-Rojas et al. [9]	✓	✓	✓			✓
Hassan et al. [30]	✓					✓
Calderon et al. [31]	✓		✓			✓
Kuvvetli [32]		✓		✓		✓
Lusiantoro et al. [33]	✓		✓	✓		✓
Our approach	✓	✓	✓	✓	✓	✓

3. Materials and methods

This section presents the materials and methods to solve the MCLPAMU problem. Section 3.1 details the mathematical formulation for the MCLPAMU, stating that Section 3.1.1 recalls the decision problem and presents the notation and parameters used. Section 3.1.2 describes the accessibility indicators. After that, Section 3.1.3 defines the decision variables based on the accessibility indicators previously defined. Finally, Section 3.1.4 presents the mixed integer programming model for MCLPAMU.

3.1. Mathematical formulation for the MCLPAMU

3.1.1. Modeling the MCLPAMU

Given a set I of locations where n facilities must be opened and a set J of demand zones, the objective consists in optimizing the coverage, the accessibility, the distances, and the disaggregation in the different demand zones using accessibility measures. We consider that each facility i has a service area previously defined (gray circles in Fig. 1); if a demand zone j is within this service area, this demand zone j is already covered. However, many demand zones j are unattended because their service area is limited. However, the unattended demand zones can reach an open facility using its mobility area previously defined (brown dash circles in Fig. 1). Using these two concepts, we can optimize different accessibility measures mentioned before. For example, regarding the distance for uncovered demand zones and the opportunities, we want the unattended demand zones to choose between open facilities. In this version of the MCLPAMU, we improved the coverage and accessibility by adding f_i mobile units $i \in I$, which increased the coverage and the number of opportunities for inhabitants of demand zones. Our approach was based on adding f_i mobile units per each opened facility. So, we supposed that each opened facility could send a mobile unit to a demand zone as long as it was unattended and within reach of a mobile unit area. As we mentioned, Fig. 1 represents an example of the MCLPAMU.

By incorporating the accessibility measures, the solution provides new information, and it is possible to study the impact of the different parameters. The new information could help the decision-maker to select the best option according to the customer's needs. Before describing the accessibility measures, it is essential to introduce and define the following sets: Set $N(i)$ represents the demand zones covered by the service area of facility $i \in I$. In contrast, the set $A(j)$ represents the facilities within the mobility area of each demand zone $j \in J$. Since we incorporated mobile units, the set $L(i)$ represents the demand zones covered by the service area of a mobile unit sent by $i \in I$. In addition,

the set $M(j)$ represents the demand zones able to reach a mobile unit allocated in the zone $j \in J$. Finally, the set $H(i)$ represents the demand zones able to locate a mobile unit of $i \in I$. A demand zone can be provided for a mobile unit as long as the demand zone is within the mobile area of any open facility $i \in I$.

3.1.2. Accessibility indicators

In this section, we describe the accessibility indicators used in this work, and Section 3.1.3 describes the decision variables associated with each one.

The Eqs. (1)–(6) represent the six accessibility indicators optimized to solve the MCLPAMU. Furthermore, we propose three new indicators (Eqs. (7) and (9)) associated with the mobile units. Each accessibility measure is defined as follows:

1. *Coverage facility*: a classic 0-1 coverage indicator to identify the demand zones directly covered by the service area of facilities.

$$y_j = \begin{cases} 1, & \text{if the demand zone } j \text{ is covered by open facility,} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Fig. 1 shows the objective of Eq. (1) visually. Notice that the indicator will take a value of one if and only if the inhabitants of the demand zone j (client symbol in color blue) are within the service area of a facility (gray circles).

2. *Service network*: a 0-1 indicator to identify the demand zones with access to the service network, either directly covered by the service area of facilities or with at least one facility within the mobility area of that demand zone.

$$a_j = \begin{cases} 1, & \text{if the demand zone } j \text{ is covered or} \\ & \text{it has at least one opportunity,} \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

We are interested that the inhabitants of demand zones j have access to a service network, meaning that they would be covered by a facility or mobile unit open or could reach facilities or open mobile units. We can observe in Fig. 1 that the indicator will be one if the inhabitants of the demand zone j are covered or they can reach a facility or mobile unit (client symbols in color blue and green, respectively). On the other hand, the indicator will be zero if the inhabitants of the demanded zone j are not covered and cannot reach a facility or a mobile open unit (client symbol in color pink).

3. *Opportunities*: an integer indicator represents the number of opportunities, i.e., the locations within the mobility area of inhabitants for each demand centroid.

$$o_j = \begin{cases} 0, & \text{if the demand zone } j \text{ is covered,} \\ |A(j) \cap M(j)|, & \text{otherwise.} \end{cases} \quad (3)$$

Eq. (3) aims to identify all the opportunities of inhabitants of a demand zone j , i.e., facilities or open mobile units within the accessibility area of j . Then, if the inhabitants of the demanded zone are covered, this indicator will be zero (client symbol in color pink in Fig. 1). On the contrary, this indicator will take an integer number, i.e., $o_j = 2$, where this number represents the opportunities that the inhabitants of the demanded zone j can reach through their mobility area (client symbol in color green in Z1 of Fig. 1).

4. *Travel cost function*: a real number indicator to represent the travel cost of inhabitants of a demanded zone to reach their opportunities (calculated by the inverse power of the distance [34]).

$$t_j = \begin{cases} 1, & \text{if the demand zone } j \text{ is covered,} \\ \frac{\sum_{i \in A(j) \text{ and } i \text{ is open}} \frac{1}{d_{ij}}}{\sum_{i \in A(j)} \frac{1}{d_{ij}}}, & \text{otherwise.} \end{cases} \quad (4)$$

To reduce the distance of the inhabitants of a demand zone j not covered, we use the inverse of the distance. This indicator is represented in Eq. (4). It will take a value of one if the inhabitants of demand zones are covered. On the other hand, this indicator will take a value between $[0, 1]$ if the inhabitants of the demand zone j are not covered. In this case, the indicator for a demand zone j will take a larger-and-better value if near opportunities are defined; i.e., low mobility is required.

5. *Distance facility*: a real number indicator representing the distance of demand zones unattended to the nearest open facility.

$$\hat{n}_j = \begin{cases} 0, & \text{if the demand zone } j \text{ is covered,} \\ \min_{i \text{ is open}} \{d_{ij}\}, & \text{otherwise.} \end{cases} \quad (5)$$

To identify the nearest facility open, we have used the indicator represented in Eq. (5). This indicator will be zero if the inhabitants of the demand zone j are covered, but it will take the value of the nearest facility open of the inhabitants of the demand zone j .

6. *Disaggregation*: a real number indicator to represent a measure of geographical disaggregation of the service network. In particular, it computes the maximal distance among all pairs of demand zones with no access to the service.

$$s_j = \begin{cases} 1, & \text{if the demand zone } j \text{ is into the service network,} \\ \frac{\min_{j' : a_{jj'}=0} \{d_{jj'}\}}{\max_{j' \in J} \{d_{jj'}\}}, & \text{if the demand zone } j \text{ is unattended.} \end{cases} \quad (6)$$

We are interested in avoiding clusters between demand zones not covered. Then, the indicator represented in Eq. (6) will be one if the inhabitants of the demand zone j have access to the service network, but it will take a real number between $[0, 1]$ when the inhabitants of demand zone j are not covered.

7. *Distance mobile unit*: a real number representing the distance of demand zones unattended to the nearest open mobile unit.

$$\bar{n}_j = \begin{cases} 0, & \text{if the demand zone } j \text{ is covered,} \\ \min_k \text{ has assigned a mobile unit } \{d_{kj}\} & \text{otherwise.} \end{cases} \quad (7)$$

Likewise, for the distance facility indicator to identify the nearest mobile unit open, we have used the indicator represented in Eq. (7). Then, this indicator will be zero if the inhabitants of the demand zone j are covered, but it will take the value of the nearest mobile unit open of the inhabitants of the demand zone j .

8. *General distance*: a real number represents the distance of the nearest open facility or mobile unit to unattended demand zones. Observe that the objective of this indicator is to identify the most

immediate opportunity (facility or mobile unit) for the inhabitants of the demand zone unattended through the indicators (5) and (7). Then, we can reduce these two indicators to only one, shown in the following Eq. (8).

$$n_j = \begin{cases} 0, & \text{if the demand zone } j \text{ is covered,} \\ \min \{ \hat{n}_j, \bar{n}_j \} & \text{otherwise.} \end{cases} \quad (8)$$

We have merged the distance facility indicator and the distance mobile unit indicator in only one indicator represented by Eq. (8). This indicator aims to identify the nearest opportunity (facility or mobile open unit). This indicator will be zero if the inhabitants of the demand zone j are covered. However, it will take the nearest opportunity (facility or mobile unit) inhabitants of the demand zone j can reach through their accessibility area.

9. *Coverage mobile unit*: a 0-1 coverage indicator to identify the demand zones covered by the service area of mobile units.

$$m_j = \begin{cases} 1, & \text{if the demand zone } j \text{ is covered by mobile unit,} \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

Ultimately, the objective of the indicator represented by Eq. (9) is the same as the coverage facility indicator but with mobile units. Notice that the indicator will take the value of one when the service area of a mobile unit covers the inhabitants of the demand zone j . On the contrary, it will take a value of zero when the service area of a mobile unit does not cover the inhabitants of the demand zone j .

Finally, the accessibility from a customer to the service area is defined as the weighted sum of all accessibility measures (Eq. (10)) where parameters β_e , ($e = 1, \dots, 6$) represent the preference of the accessibility measures. Notice that the indicators and variables $\beta_1 \dots \beta_6$ are criteria and weights in multicriteria optimization. On the other hand, Section 5.3 details the computation of values of $\beta_1 \dots \beta_6$. The MCLPAMU determines the n facilities to be opened among the set I and locates f_i mobile units among a subset of open facilities. Through Eq. (10), we are improving the coverage, accessibility, opportunities, and distances.

$$z = \sum_{j \in J} (\beta_1 a_j + \beta_2 (y_j + m_j) + \beta_3 t_j - \beta_4 n_j + \beta_5 o_j + \beta_6 s_j). \quad (10)$$

3.1.3. Decision variables

The decision variables used in the MCLPAMU are divided into four groups of binary variables. The first two sets are associated with the facilities, and the last two are with the mobile units.

$$z_i = \begin{cases} 1, & \text{if the facility } i \in I \text{ is opened,} \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

$$v_{ij} = \begin{cases} 1, & \text{if the demand zone } j \in J \text{ is not covered and} \\ & i \in I \text{ is the nearest open facility,} \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

$$q_{ij} = \begin{cases} 1, & \text{if the facility } i \in I \text{ locate a mobile unit} \\ & \text{in the demand zone } j \in J, \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

$$u_{ijk} = \begin{cases} 1, & \text{if the demand zone } j \in J \text{ is not covered and} \\ & i \in I \text{ has located a mobile unit in} \\ & \text{the demand zone } k \in J \text{ and it is the nearest,} \\ 0, & \text{otherwise.} \end{cases} \quad (14)$$

Notice that the accessibility indicators defined in the previous section are decision variables too. Therefore, we define the accessibility indicators as the following decision variables: $y_j \in \{0, 1\}$, $a_j \in \{0, 1\}$, $o_j \in \mathbb{Z}_0^+$, $t_j \in [0, 1]$, $n_j \geq 0$, $s_j \geq 0$, $\hat{n}_j \geq 0$, $\bar{n}_j \geq 0$ and $m_j \in \{0, 1\}$.

3.1.4. MILP for the MCLPAMU

Now, we present the mixed-integer linear program for the MCLPAMU.

$$\max z = \frac{1}{|J|} \sum_{j \in J} \left(\beta_1 a_j + \beta_2 (y_j + m_j) + \beta_3 t_j + \beta_4 n_j + \beta_5 \frac{o_j}{|A(j)| + |M(j)|} + \beta_6 s_j \right) \quad (15)$$

subject to:

$$\sum_{i \in I} z_i = n \quad (16)$$

$$\sum_{i:j \in N(i)} z_i \leq |N(i)| y_j \quad \forall j \in J \quad (17)$$

$$y_j \leq \sum_{i:j \in N(i)} z_i \quad \forall j \in J \quad (18)$$

$$v_{ij} \leq z_i \quad \forall i \in I, \forall j \in J \quad (19)$$

$$\sum_{i \in I} v_{ij} = 1 - y_j - m_j \quad \forall j \in J \quad (20)$$

$$\sum_{i:j \in L(i)} q_{ij} \leq |L(i)| m_j \quad \forall j \in J \quad (21)$$

$$m_j \leq \sum_{i:j \in L(i)} q_{ij} \quad \forall j \in J \quad (22)$$

$$\sum_{j \in J} q_{ij} \leq f_i z_i \quad \forall i \in I \quad (23)$$

$$\sum_{i \in I} q_{ij} \leq 1 \quad \forall j \in J \quad (24)$$

$$u_{i,j} \leq q_{i,j} \quad \forall i \in I \quad (25)$$

$$\sum_{i \in I} \sum_{k \in H(i)} u_{ijk} \leq 1 - y_j - m_j \quad \forall j \in J, j \neq k \quad (26)$$

$$m_j \leq 1 - y_j \quad \forall j \in J \quad (27)$$

$$a_j \leq y_j + m_j + \sum_{i \in A(j)} v_{ij} + \sum_{i \in I} \sum_{k \in M(k)} u_{ijk} \quad \forall j \in J, j \neq k \quad (28)$$

$$t_j \leq \frac{\sum_{i \in A(j)} \frac{z_i}{d_{ij}}}{\sum_{i \in A(j)} \frac{1}{d_{ij}}} + y_j + m_j \quad \forall j \in J \quad (29)$$

$$\hat{n}_j = \left(\frac{\max_{i \in I} \{d_{ij}\} - \sum_{i \in I} v_{ij} d_{ij}}{\max_{i \in I} \{d_{ij}\}} \right) \quad \forall j \in J \quad (30)$$

$$\bar{n}_j = \left(\frac{\max_{k \in J} \{d_{jk}\} - \sum_{i \in I} \sum_{k \in J} u_{ijk} d_{jk}}{\max_{k \in J} \{d_{jk}\}} \right) \quad \forall j \in J \quad (31)$$

$$n_j \leq \hat{n}_j \quad \forall j \in J \quad (32)$$

$$n_j \leq \bar{n}_j \quad \forall j \in J \quad (33)$$

$$o_j \leq \sum_{i \in A(j)} z_i + \sum_{i:j \in M(j)} q_{ij} \quad \forall j \in J \quad (34)$$

$$o_j \leq (|A(j)| + |M(j)|) (1 - y_j - m_j) \quad \forall j \in J \quad (35)$$

$$s_j \leq \frac{d_{jj'}}{\max_{j' \in J} \{d_{jj'}\}} + (a_j + a'_j) \quad \forall j \in J, \forall j' \in J : j' \neq j \quad (36)$$

$$z_j, y_j, v_{ij}, a_j, m_j, q_{ij}, u_{ijk} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall k \in J \quad (37)$$

$$t_j, n_j, \hat{n}_j, \bar{n}_j, s_j \geq 0 \quad \forall j \in J \quad (38)$$

$$o_j \in \mathbb{Z}_0^+ \quad \forall j \in J \quad (39)$$

The objective function (15) maximizes a weighted sum of the normalization of accessibility indicators. In particular, β_i parameters represent the weights, where $\beta_i \in [0, 1]$ and $i = 1, \dots, 6$. Constraints (16) ensures to open n facilities. The constraints (17) and (18) guarantee that the demand zones are covered for open facilities. The constraints (19) and (20) allow the demand zones not covered to have only one nearest opportunity. The inequalities (21) and (22) guarantee that open mobile units' demand zones are covered. Constraint (23) ensures open at most f_i mobile units for each open facility i . The constraints (24) state that a mobile unit can be present in only one demand zone. On the other hand, constraints (25) guarantee that the demand zones can only access open mobile units while the constraints (26) state that the not covered demand zones can access only a mobile unit. Constraints (27) ensure that if a facility covers a demand zone, any mobile unit must not cover this demand zone. Constraints (28) determine if a demand zone is in the service network. A demand zone is within the service network when it is covered by an open facility or mobile unit or has at least one opportunity to reach an open facility or mobile unit. Constraints (29) define the value of the travel cost for the demand zone. The idea is to open near opportunities (facilities and/or mobile units) for the unmet demand zones. Constraints (30) state the nearest open facility to a demand zone not covered by the service network. Likewise, constraints (31) state that the nearest open mobile unit to a demand zone is not covered. Using constraints (32) and (33), it is possible to identify the nearest opportunity (facility or mobile unit) for a demand zone not covered. Constraints (34) and (35) state the number of opportunities per each not-covered demand zone. At the same time, the constraints (36) establish if the demand zone is in the service network. Eqs. (37) - (39) represent the nature of the decision variables.

The MCLPAMU belongs to the NP-hard complexity class since its decision version can be reduced to the decision version of the MCLPA. In [35], the authors showed that commercial solvers for the MCLPA formulation could solve large instances in a reasonable amount of computational time. However, for the MCLPAMU formulation, commercial solvers can only solve small and medium instances (see Section 5) since the MCLPAMU additionally considers the mobile unit's assignment.

To obtain feasible high-quality solutions for large instances, we have combined a metaheuristic with a modification of the MILP model explained in Section 3.1.4. Recalls that the combination of exact and heuristic methods is named matheuristic. The following Section 4 details the matheuristic implemented for the MCLPAMU.

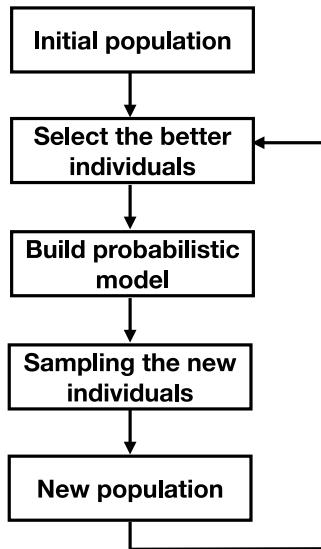


Fig. 2. General EDA.

4. Matheuristic algorithm for the MCLPAMU

The *matheuristics* are methods that embed mathematical programming inside a heuristic or metaheuristic framework. These methods have been applied to several combinatorial optimization problems, e.g., the vehicle routing problem [36,37], the facility location problem [38], and the set covering problem [39]. We propose a matheuristic approach to solve the MCLPAMU based on the mathematical formulation presented in Section 3.1 and a metaheuristic named Estimation of Distribution Algorithm (EDA). The EDA decides which facilities should be open, i.e., the EDA determines the values of the vector z . We can compute the demand zones covered by a feasible vector \bar{z} by the EDA. These demand zones are stored in the vector \bar{y} , and they are used to solve a modified formulation of MCLPAMU that we have named MCLPAMU(\bar{z}, \bar{y}). The next Section 4.1 details the metaheuristic EDA used in the matheuristic and the Section 4.3.1 describes the MCLPAMU(\bar{z}, \bar{y}) formulation. Finally, Section 4.4 describes the *matheurisitic* implemented.

4.1. Estimation of distribution algorithms

The estimation of distribution algorithms are stochastic optimization techniques that explore the space of potential solutions by building and sampling explicit probabilistic models of promising candidate solutions [40]. The EDAs emerge due to the evolutionary computation algorithms like *Genetic Algorithms* (GAs), which depend on several parameters associated with them (crossover and mutation operators, probabilities of crossover and mutation, size of the population, rate of generational reproduction, number of generations, etc.). Since GAs depends on several parameters and the fact that a person with no experience selects a good combination of all these parameters motivated to create the EDAs [41]. The EDAs start with a solutions group (also known as population), each solution is named an individual, and these solutions are encoded. Fig. 2 shows the general steps of an EDA.

The EDAs are classified by the probabilistic model used to estimate the probability distribution and sample the new solutions. The probabilistic models usually used are the no dependency, the pairwise dependencies, the multivariate dependencies, and the full dependence model [42].

4.2. EDA for the MCLPA and MCLPAMU

For the MCLPA and MCLPAMU, we use an EDA without dependencies known as the Univariate Marginal Distribution Algorithm (UMDA). The UMDA was introduced by [43,44]. The UMDA is the most basic EDA developed for binary optimization problems. The UMDA decomposes the probability of a candidate solution (z_1, z_2, \dots, z_n) into the product of probabilities of individual variables as:

$$p(z_1, z_2, \dots, z_n) = p(z_1)p(z_2), \dots, p(z_n) = \prod_{i=1}^{|I|} p(z_i), \quad (40)$$

where $p(z_i)$ is the probability of variable z_i , and $p(z_1, z_2, \dots, z_n)$ is the probability of the candidate solution (z_1, z_2, \dots, z_n) . Suppose then we have $|I|$ variables in a problem. Using a UMDA approach, we have $|I|$ probability distributions (one per variable), and each of these distributions defines the probabilities of the different values for the corresponding variable. Each probability is stored into the probability vector $p = (p_1, p_2, \dots, p_i)$ as the probabilistic model, where p_i denotes the probability of a 1 in position i of solution string. To generate the probability vector, the selected population through the fitness function is used, e.g., to create a new binary string from the probability vector, for each position i , a 1 is generated in this position with probability p_i , for instance, if $p_1 = 0.5$, we generate a 1 in the first position of a new candidate solution with the probability of 50%. In the UMDA, each variable is independently generated based on the entries in the probability vector p_i per iteration (or generation). The general form of UMDA is shown in the Algorithm 1.

Algorithm 1 EDA-UMDA Pseudocode

```

1:  $t := 0$ 
2:  $D_t \leftarrow$  Generate population initial with  $N$  individuals.
3: while the stop criteria is not reached do
4:   Using fitness function ranks the  $N$  individuals of the population  $D_t$ 
5:    $S_t \leftarrow$  Select the superior  $Se$  solutions ( $Se < N$ ) from the population  $D_t$  ranked.
6:    $p_t(z|S_t) \leftarrow$  Calculate the probability vector using  $S_t$ 
7:    $D_{t+1} \leftarrow$  Sample  $N$  individuals using the probability vector  $p_t(z|S_t)$  to create the new population.
8:    $t := t + 1$ 
9: end while
  
```

Usually, the metaheuristics are composed of a representation and a fitness function. The representation encodes the individuals for better manipulation, while the fitness function ranks the solutions by identifying their quality. The representation and the fitness function are described below.

4.2.1. Representation

The representation is an important step during the design of any metaheuristic. Our proposed EDA uses a binary representation for the MCLPAMU(\bar{z}, \bar{y}).

We recall that the set I is used to represent the facilities. On the other hand, the parameter n indicates the number of facilities that must be open. Then, we use a binary vector \bar{z} of size $|I|$ to represent a candidate solution, where $\bar{z}_i = 1$ indicates that a facility is open at $i \in I$. Since we use a random generation of vectors \bar{z} , we do not guarantee to use exactly n locations; thus, a candidate solution may be infeasible. We used a penalty method while designing the fitness function and a repair method for infeasibilities. Notice that for each vector \bar{z} feasible, we can compute the vector \bar{y} associated with it, using the set $N(i)$ mentioned in Section 3.1.1.

4.2.2. Fitness function

The fitness function ranks each candidate solution to identify the best solution. We have used two methods to rank the candidate solutions in this case. We have implemented a penalty function method to identify feasible solutions and a repair solution method to maintain always feasible solutions in each iteration. The Eq. (41) represents the penalty function:

$$P(\bar{z}) = \begin{cases} 0, & \text{if } \sum_{i \in I} \bar{z}_i = n \\ -(\mathcal{M} - \sum_{i \in I} z_i), & \text{if } \sum_{i \in I} \bar{z}_i \neq n \end{cases} \quad (41)$$

where \mathcal{M} is a large enough number. Once the candidate solution is classified as feasible, we use the MCLPAMU(\bar{z}, \bar{y}) formulation (See Section 4.3.1) to rank it. Notice that we use the MCLPAMU(\bar{z}, \bar{y}) formulation to rank a solution if only if it is feasible, the objective is to maximize Eq. (42).

$$f(\bar{z}) = \frac{1}{|J|} \sum_{j \in J} \left(\beta_1 a_j + \beta_2 (y_j + m_j) + \beta_3 t_j + \beta_4 n_j + \beta_5 \frac{o_j}{|A(j)| + |M(j)|} + \beta_6 s_j \right) \quad (42)$$

Each candidate solution (feasible or not) will take the value of Eq. (43), where the best solution maximizes that value. Notice that given a feasible solution \bar{z} , the matheuristic can compute the value for the rest of the variables solving MCLPAMU(\bar{z}, \bar{y}) formulation.

$$f'(\bar{z}) = f(\bar{z}) + P(\bar{z}) \quad (43)$$

4.3. Infeasibilities repair method

We implement a repair method for the infeasible candidate solutions, in particular, a candidate solution \bar{z} is infeasible when $\sum_{i \in I} \bar{z}_i > n$ or $\sum_{i \in I} \bar{z}_i < n$. The repair method identifies the positions with 1 or 0, respectively, for each case. After that, the repair method randomly selects a position within \bar{z} to change the value for 1 to 0 or vice versa, according to the case. The Algorithm 2 details the repair method implemented.

Observe that using the repair method, the entire population will be feasible in each iteration of the matheuristic. Similarly to the penalty function method, each candidate solution is ranked using the MCLPAMU (\bar{z}, \bar{y}) formulation described in the next Section 4.3.1 likewise the penalty function method the objective is to maximize Eq. (42).

4.3.1. MCLPAMU(\bar{z}, \bar{y}) parameterized

The matheuristic is implemented formulated the mathematical formulation for the MCLPAMU (Section 3.1) considering the vectors computed by the EDA: \bar{z} and \bar{y} . Notice that given a feasible solution \bar{z} , we can compute \bar{y} . For this new formulation, we substitute the Constraints (16)–(18) and Constraints (21)–(22) with Constraints (44) and (45). The rest of the mathematical formulation is the same.

$$z_i = \begin{cases} 1, & \text{if } \bar{z}_i = 1 \text{ then use } z_i \geq \bar{z}_i \quad \forall i \in I \\ 0, & \text{if } \bar{z}_i = 0 \text{ then use } z_i \leq \bar{z}_i \quad \forall i \in I \end{cases} \quad (44)$$

$$y_j = \begin{cases} 1, & \text{if } \bar{y}_j = 1 \text{ then use } y_j \geq \bar{y}_j \quad \forall j \in J \\ 0, & \text{if } \bar{y}_j = 0 \text{ then use } y_j \leq \bar{y}_j \quad \forall j \in J \end{cases} \quad (45)$$

The MCLPAMU(\bar{z}, \bar{y}) formulation used in the matheuristic is presented as follows:

max Eq. (15)

subject to:

Constraints (19)–(20)

Constraints (23)–(38)

Constraints (44)–(45)

Notice that we use the vectors \bar{z} and \bar{y} to define the values of z and y , respectively, in the new MCLPAMU formulation. In other words,

Algorithm 2 Repair method Pseudocode

Input:

$n \leftarrow$ Number of facilities must be opened.
 $\bar{z} \leftarrow$ Candidate solution infeasible.

Output:

$\bar{z} \leftarrow$ Candidate solution feasible.

```

1:  $h \leftarrow \text{CountOnes}(\bar{z})$ 
2:  $c \leftarrow n - h$ 
3: if  $c > 0$  then
4:   while  $c > 0$  do
5:      $\bar{a} \leftarrow \text{IdentifyPositionsWithZero}(\bar{z})$ 
6:      $\bar{u} \leftarrow \text{ToDoFalsePositionsWithZero}(\bar{a})$ 
7:      $p_1 \leftarrow \text{SelectRandomPositionNotUsed}(\bar{a}, \bar{u})$ 
8:      $p_2 \leftarrow \text{SelectRandomPositionNotUsed}(\bar{a}, \bar{u})$ 
9:      $p \leftarrow \text{SelectRandomPosition}(p_1, p_2)$ 
10:     $\bar{z}_p \leftarrow 1$ 
11:     $u_p \leftarrow \text{True}$ 
12:     $c \rightarrow c - 1$ 
13:  end while
14: else
15:   while  $c < 0$  do
16:      $\bar{a} \leftarrow \text{IdentifyPositionsWithOnes}(\bar{z})$ 
17:      $\bar{u} \leftarrow \text{ToDoFalsePositionsWithOnes}(\bar{a})$ 
18:      $p_1 \leftarrow \text{SelectRandomPositionNotUsed}(\bar{a}, \bar{u})$ 
19:      $p_2 \leftarrow \text{SelectRandomPositionNotUsed}(\bar{a}, \bar{u})$ 
20:      $p \leftarrow \text{SelectRandomPosition}(p_1, p_2)$ 
21:      $\bar{z}_p \leftarrow 0$ 
22:      $u_p \leftarrow \text{True}$ 
23:      $c \rightarrow c + 1$ 
24:   end while
25: end if
26: return  $\bar{z}$ 
```

we have established the values of the decision variables represented by the Eqs. (11) and (12) defining lower and upper bounds. The next Section 4.4 details the matheuristic implemented.

4.4. Matheuristic for the MCLPAMU

The EDA used in the matheuristic follows the same structure as Algorithm 1, considering the mathematical model described in Section 4.3.1. Notice that we have used the EDA to decide which facilities must be open. Using the facilities opened (\bar{z}), we can compute the demand zones covered by them and store in \bar{y} . We are using this information (\bar{z} and \bar{y}) to compute where to allocate the mobile units and the rest of the decision variables through the formulation described in Section 4.3.1. Thus, given an initial probability $p_0(\bar{z})$ to generate the first EDA population with size M , each individual \bar{z} is ranked by using the MCLPAMU (\bar{z}, \bar{y}) formulation described before. The EDA uses a predefined selection method from the ranked population to choose the better Se individuals (We have used a truncate method). After that, the EDA estimates the probabilities using the Se individuals selected and samples the new generation with size M . The process ends when a maximum of iterations T is reached. Algorithm 3 describes the matheuristic method.

In this section, we described the matheuristic proposed, which is composed of an Estimation of the Distribution Algorithm and a parameterized MCLPAMU formulation. Section 5 describes the experimental results obtained for the MCLPAMU formulation and the matheuristic. Moreover, we detail the instances proposed where we have considered the COVID-19 situation in Mexico. On the other hand, we described the computation of the weights used in the objective function and the parameter set used in the matheuristic.

Algorithm 3 Matheuristic Pseudocode

Input:

- $M \leftarrow$ Size of population.
- $Se \leftarrow$ Number of individuals to select.
- $p_0(\bar{z}) \leftarrow$ Probability vector initial.
- $T \leftarrow$ Maximum number of iterations.

Output:

- $E \leftarrow$ The best solution in all iterations.

- 1: $t \leftarrow 0$
- 2: $D_t \leftarrow \text{GenerateInitialPopulation}(M, p_0(\bar{z}))$.
- 3: **while** $t < T$ **do**
- 4: $\bar{Y}_t \leftarrow$ Compute the vector \bar{y} for each individual in the population D_t
- 5: $\text{RankPopulationSolvingMCLPAMU}(D_t, Y_t)$
- 6: $E \leftarrow \text{SaveBestSolution}(D_t)$
- 7: $S_t \leftarrow \text{SelectSuperiorSolutions}(D_t, Se)$
- 8: $p_{t+1}(\bar{z}) \leftarrow \text{EstimateProbabilities}(S_t)$
- 9: $D_{t+1} \leftarrow \text{SampleNewPopulation}(M, p_{t+1}(\bar{z}))$
- 10: $t \leftarrow t + 1$
- 11: **end while**
- 12: **return** E

5. Experimental results

This section presents the experimental results for the methodologies described in Section 3. Section 5.1 gives a short description of the generated instances, which take the situation of COVID-19 in Mexico and illustrate the benefits of the MCLPAMU formulation. Section 5.2 shows the parameter settings for the EDA-MCLPA and the matheuristic. Section 5.3 describes the compute of the weights ($\beta_1 \dots \beta_6$) for the objective function. Section 5.4 details the numerical results obtained in the different methods presented in this work. We compare the MCLPA against the EDA-MCLPA (Section 5.4.1) and the MCLPAMU versus the matheuristic (Section 5.4.2). We have noticed an improvement in the experimental stage using the MCLPAMU formulation.

5.1. Instances generated for COVID-19 case in Mexico

In this section, we detail the creation of the instances set, which is based on data from official Mexican databases available for anyone interested and contains data from the COVID-19 situation in Mexico that we explained in the following. At the beginning of the COVID-19 pandemic, and considering that COVID-19 tests were scarce, the Mexican government established that only a few hospitals could receive COVID-19 tests, which were called COVID-19 Hospitals (C19H). There is no official database (DB) where you can find the C19H. However, in the App COVID19MX¹ you can find the C19H per municipality,² where it is possible to find 541 C19H.³ With the locations of each zone, it is possible to calculate the distance between municipalities and C19Hs considering the latitude and longitude of each one, and considering the Haversine formula with the distance got by Google Maps [45].

The number of COVID-19 tests in Mexico was not available as public information. Therefore, these data were estimated with the news of several local newspapers.⁴ Through the number estimated of COVID-19 tests; it was possible to compute the number n of hospitals that must be open. Thus, the set I of C19H, the set J of municipalities, and the number n of hospitals were already known. The service area for each

Table 2
Class of instances.

Class	ID	Region		$ I $	$ J $	n
		Name	ID-R			
A	1	Aguascalientes	1	4	11	2
	2	Baja California	2	11	5	5
	3	Baja California Sur	3	14	5	1
	4	Campeche	4	4	11	1
	5	Coahuila	5	17	38	5
	6	Colima	6	7	10	2
	7	Chiapas	7	11	118	4
	8	Chihuahua	8	18	67	6
	9	Ciudad de Mexico	9	36	16	17
	10	Durango	10	4	39	3
	11	Guanajuato	11	50	46	7
	12	Guerrero	12	12	81	4
	13	Hidalgo	13	25	84	2
	14	Jalisco	14	41	125	11
	15	Mexico	15	38	125	19
	16	Michoacan	16	34	113	5
	17	Morelos	17	7	33	3
	18	Nayarit	18	3	20	2
	19	Nuevo Leon	19	8	51	8
	20	Oaxaca	20	36	570	4
	21	Puebla	21	16	217	7
	22	Queretaro	22	4	18	2
	23	Quintana Roo	23	10	10	3
	24	San Luis Potosi	24	16	58	4
	25	Sinaloa	25	19	18	4
	26	Sonora	26	20	72	5
	27	Tabasco	27	7	17	2
	28	Tamaulipas	28	29	43	6
	29	Tlaxcala	29	3	60	2
	30	Veracruz	30	23	212	8
	31	Yucatan	31	8	106	4
	32	Zacatecas	32	6	58	2
B	33	North	2,3,5,8,10,19,24,25,26,28	156	396	47
	34	West	1,6,11,14,16,18,22,32	149	401	33
	35	Center	9,12,13,15,17,21,29	137	616	54
	36	South	4,7,20,23,27,30,31	99	1044	26
C	37	Mexico	1, ..., 32	541	2457	160

hospital $i \in I$ was set to 50 km, and each hospital could send a mobile unit f_i at most 80 km. The service area for a mobile unit f_i was 12.5 km, and the accessibility area for each municipality was 25 km. Considering the previous information, three instance sets were generated: Class A, B, and C. Class A contained 32 instances, and Mexico was divided into states. Class B comprised four instances where Mexico was divided into economic zones (North, West, Center, and South). Class C contained just one instance that considered all the municipalities of Mexico.

Table 2 shows a total of 37 instances. Column 1 presents the instance class, Column 2 describes the ID of the instance, and Columns 3 and 4 show the region that includes the name and the ID of the region (ID-R). Notice for instances of Class B and C; the ID-R is taken from the ID-R of Class A, e.g., the ID-R = 2, 3, and 5 means the instance includes the region of Baja California, Baja California Sur, and Coahuila. Columns 5, 6, and 7 describe the total of hospitals, municipalities, and the number of hospitals that must be open, respectively. The number of mobile hospitals f_i has been generated randomly between 1 and 3 for each hospital $i \in I$. The instances set can be downloaded in the following link: <https://doi.org/10.6084/m9.figshare.20460867.v3>.

5.2. Parameter settings

Before starting with the numerical results, we performed a test to select the best parameters configuration. The test was done with the irace package. The irace package takes advantage of the iterated racing procedure, and it has been used in many different metaheuristics and matheuristics studies to select the best parameters set [see 46–48]. We have used the irace package version 3.4.1 developed by [49]. Broadly,

¹ COVID19MX App: <https://play.google.com/store/apps/details?id=mx.gob.www>

² Mexico is divided into 32 states, including 2,457 municipalities.

³ C19H DB: <https://doi.org/10.6084/m9.figshare.20460867.v3>

⁴ El financiero: <https://bit.ly/3LHYNsh>

Table 3
Parameters used in MCLPA and MCLPAMU.

Tests	p_0	M	Se	T
P_1	0.16	883	147	10
P_2	0.24	48	5	6

irace starts selecting a budget of experiments and instances set, where we selected 500 experiments and instances 20,33,34,35, and 36. After that, we selected the range of the different parameters in this case: $p_0 \in [0.10, 0.60]$, $M \in [100, 1000]$, $Se \in [75, 850]$ and $T \in [5, 10]$. In each iteration, irace selects the elites configurations, after several iterations the irace package provides the best parameter configuration. We did two different tests (P_1 & P_2), and the parameters set found are shown in Table 3. In the case of the test P_2 , we selected the same number of experiments, and the same instances set likewise test P_1 . However, we used the following ranges: $p_0 \in [0.10, 0.30]$, $M \in [10, 60]$, $Se \in [5, 55]$ and $T \in [4, 8]$.

5.3. Compute the weights used in the objective function

To compute the weights $\beta_1 \dots \beta_6$, we considered two main indicators: The rate of death people by COVID-19 (δ_j) and the poverty index (γ_j) both per municipality $j \in J$. In Mexico, the government publishes daily a database with the number of people infected and died from COVID-19. We cleaned and categorized this DB to obtain the number of dead people (d_j) per municipality $j \in J$. The DB is available on the Mexican government's official web page.⁵ Besides, we obtained the value of the population (p_j) in each municipality $j \in J$; this information was obtained by INEGI.⁶ Using this information, we obtained a rate $\delta_j = d_j/p_j$ per municipality j in each state of Mexico where $\delta_j \in [0, 1]$. On the other hand, through the National Council for the Evaluation of Social Development Policy (CONEVAL in Spanish), we obtained the poverty index ($\gamma_j \in [0, 1]$) per municipality $j \in J$ in its official web page.⁷ There are many ways to compute the weights; the objective function will be sensitive to them. For example, in [31] prioritize socio-economic and epidemiological indicators per each demand zone. We have computed the weights using the indicator mentioned before (γ_j) since according to United Nations Mexico⁸ the people in poverty situation have the double risk to die than the rest of population. Therefore, using these two main indicators (δ_j and γ_j), the first weight of each municipality j was computed by

$$\beta_{1j} = (\delta_j \times \theta) + \gamma_j, \quad \forall j \in J \quad (46)$$

where θ is a large enough number that prioritizes the municipalities j with more deaths and avoids loss of precision in decimal numbers; in this case, we have used $\theta = 1000$. Once computed the β_{1j} for each j , we normalize dividing by the sum of β_{1j} , that means $\beta_{1j} = \frac{\beta_{1j}}{\sum_{j \in J} \beta_{1j}}$ for each municipality j . The rest of the weights $\beta_{2j} = \frac{\beta_{1j}}{2}, \beta_{3j} = \frac{\beta_{2j}}{2} \dots \beta_{6j} = \frac{\beta_{5j}}{2}$, are computed divided by two. Therefore, β_{1j} is the weight most important, following for β_{2j} and consecutively. We have divided the values by 2 since a sensitivity analysis was done by Ibarra-Rojas et al. [9] shows that dividing by 2 assures the maximization of accessibility measures. Observe that using Eq. (46) the value of $\beta_{1j} \in [0, 0.5]$. Therefore, to obtain $\sum_{i=1}^6 \sum_{j=1}^{|J|} \beta_{ij} = 1$ is necessary, divide

⁵ COVID-19 open data: <https://www.gob.mx/salud/documentos/datos-abiertos-152127>

⁶ INEGI open data: https://www.inegi.org.mx/programas/ccpv/2010/#Datos_abiertos

⁷ Poverty Index by CONEVAL: https://www.coneval.org.mx/Medicion/MP/Paginas/Pobreza_2020.aspx

⁸ United Nations Mexico: <https://coronavirus.onu.org.mx/en-mexico-el-covid-19-mata-al-doble-a-los-pobres-e-indigenas>

B_{ij} by $|J|$ when $i \geq 2$. We select these values in the order shown in Section 3.1.4 because, in the work of Ibarra-Rojas et al. [9], a sensitivity analysis showed these values obtained the best results. Recall that there are many methods to compute the weights. For example, Rezaei [50] proposed an integer linear model based on criteria previously decided by the decision-maker to compute the weights. We selected the approach mentioned in this section because we are interested in reducing the deaths of low-income people for COVID-19.

5.4. Numerical results

The computational experiments were performed on a Mac Pro with a 3.5 GHz Intel Xeon E5, running Monterey operating system, and 32 GB of RAM. The MCLPA and MCLPAMU formulations, the EDA-MCLPA, and the matheuristic were implemented in C/C++. In the case of MCLPA, MCLPAMU, and matheuristic we have used CPLEX 12.10 with a time limit of 21,600 seconds. The EDA-MCLPA and the matheuristic were replicated ten times and one time with the parameter set P_1 , respectively. Using the repair method explained in Section 4.3 and the parameter set P_2 , the EDA-MCLPA and the matheuristic were replicated ten times.

5.4.1. Comparison between MCLPA and EDA-MCLPA

This section shows in Table 4 the results obtained with the MCLPA formulation and the EDA-MCLPA. Column 1 displays the instance name, and columns 2 and 3 display the results of the MCLPA formulation. Column 2 shows the Service Network percentage, and column 3 displays the time in seconds to solve the MCLPA formulation. Column 4 to column 7 display the results of the EDA-MCLPA. Column 4 shows the average of the Service Network percentage, column 5 displays the average time in seconds, and column 6 shows the average GAP.⁹ Finally, column 7 displays the best Service Network percentage found in the EDA-MCLPA.

All instances have been solved until they reach the optimal MCLPA formulation solution. Besides, the computational times were reasonable in all of them. On the other hand, the EDA-MCLPA obtained GAPs less than 17% without repair and less than 12% with repair in all the instances with short computational times. The EDA-MCLPA, in some instances, obtained a better average Service Network percentage than the MCLPA formulation, for example, Instance 11 and 21. However, the result of the objective function in MCLPA formulation was larger than the EDA-MCLPA. The complete results are in <https://doi.org/10.6084/m9.figshare.20460867.v3>.

5.4.2. Comparison between MCLPAMU and matheuristic

This section shows the results of MCLPAMU formulation and the matheuristic in Table 5. Column 1 displays the instance ID; columns 2 and 3 show the results of MCLPAMU formulation. Column 2 displays the Service Network percentage, and column 3 shows the computational time. Meanwhile, columns 4, 5 and 6 display the results of the matheuristic using the parameters set P_1 . Observe that set is the same that we used in the EDA-MCLPA. Column 4 displays the Service Network Percentage obtained with parameter set P_1 , column 5 exhibits the computational time, and column 6 shows the GAP computed. On the other hand, columns 7 to 10 display the results of the matheuristic using parameters set P_2 . Column 7 shows the Service Network Percentage Average, while columns 8 and 9 show the average time and the average GAP, respectively. The final column 10 shows the best service network percentage found in all the iterations.

Instances 35, 36 and 37 could not be solved with the MCLPAMU formulation; the remaining instances have been solved to reach the

⁹ In general, the gap is computed by $GAP = \frac{|ExactMethodSolution - BestSolutionEDA|}{ExactMethodSolution} \times 100$

Table 4

Results of the MCLPA formulation and the EDA-MCLPA.

Instance	MCLPA		EDA-MCLPA without repair				EDA-MCLPA with repair			
	SN (%)	Time (s)	Avg SN (%)	Avg time (s)	Avg GAP (%)	Best SN (%)	Avg SN (%)	Avg time (s)	Avg GAP (%)	Best SN (%)
1	100.00	0.01	100.00	0.05	0.00	100.00	100.00	0.00	0.00	100.00
2	100.00	0.01	100.00	0.05	0.00	100.00	100.00	0.01	0.00	100.00
3	20.00	0.01	20.00	0.05	0.00	20.00	20.00	0.00	0.00	20.00
4	18.18	0.00	18.18	0.05	0.00	18.18	18.18	0.05	0.00	18.18
5	76.32	0.04	76.05	0.06	0.06	76.32	75.53	0.05	0.24	76.32
6	100.00	0.01	100.00	0.06	0.00	100.00	100.00	0.05	0.00	100.00
7	63.56	0.18	63.56	0.18	0.00	63.56	63.47	0.14	0.15	63.56
8	38.81	0.07	38.81	0.08	0.00	38.81	38.81	0.08	0.00	38.81
9	100.00	0.01	100.00	0.06	0.00	100.00	100.00	0.07	0.00	100.00
10	12.82	0.01	12.82	0.07	0.00	12.82	12.82	0.01	0.00	12.82
11	97.83	0.21	96.96	0.09	0.97	97.83	97.61	0.03	0.29	97.83
12	43.21	0.08	43.21	0.10	0.00	43.21	43.21	0.02	0.00	43.21
13	64.29	2.80	64.29	0.19	0.00	64.29	63.33	0.02	0.56	64.29
14	88.80	0.79	85.36	0.17	4.48	87.20	87.04	0.05	2.86	88.00
15	100.00	1.30	100.00	0.15	0.02	100.00	100.00	0.05	0.00	100.00
16	77.88	0.58	77.35	0.17	0.22	77.88	76.81	0.04	1.16	77.88
17	100.00	0.03	100.00	0.07	0.00	100.00	100.00	0.01	0.00	100.00
18	35.00	0.01	35.00	0.06	0.00	35.00	35.00	0.01	0.00	35.00
19	43.14	0.00	43.14	0.10	0.00	43.14	43.14	0.01	0.00	43.14
20	55.09	12.98	54.25	1.74	1.91	55.09	53.93	0.38	2.39	55.09
21	91.71	0.85	91.80	0.30	0.03	92.63	91.52	0.08	0.16	92.63
22	55.56	0.01	55.56	0.06	0.00	55.56	55.56	0.01	0.00	55.56
23	60.00	0.01	60.00	0.05	0.00	60.00	60.00	0.01	0.00	60.00
24	67.24	0.07	67.24	0.08	0.00	67.24	66.21	0.02	0.13	67.24
25	61.11	0.02	61.11	0.06	0.02	61.11	59.44	0.01	1.01	61.11
26	22.22	0.07	22.22	0.09	0.00	22.22	22.22	0.03	0.28	22.22
27	82.35	0.01	82.35	0.06	0.03	82.35	81.18	0.01	0.38	82.35
28	55.81	0.07	55.35	0.07	0.14	55.81	53.95	0.02	0.64	55.81
29	98.33	0.01	98.33	0.12	0.00	98.33	98.33	0.01	0.00	98.33
30	79.72	1.01	79.48	0.28	0.08	79.72	79.62	0.16	0.03	79.72
31	72.64	0.12	72.64	0.22	0.00	72.64	72.64	0.02	0.00	72.64
32	27.59	0.03	27.59	0.10	0.00	27.59	27.59	0.01	0.00	27.59
33	49.49	10.96	38.61	0.88	13.31	40.40	40.88	0.77	8.42	42.93
34	86.53	15.16	70.62	1.05	11.90	73.82	75.86	0.55	6.15	79.80
35	91.23	47.24	88.02	1.15	2.06	89.61	90.99	0.90	0.14	91.23
36	73.75	56.18	58.32	2.97	16.42	67.24	64.36	1.94	7.29	68.49
37	84.53	969.60	64.20	16.70	17.84	67.89	69.36	11.98	11.29	73.59

optimal solution. The computational times using MCLPAMU formulation were reasonable except for larger instances. The implemented matheuristic solves all the instances except instance 37. Nevertheless, using the parameter set P_1 , the computational time increased exponentially because there were many individuals (883) and MCLPAMU (\bar{z}, \bar{y}) formulation inside the matheuristic solved each feasible individual. However, the GAPs obtained were less than 8% using this parameter set. On the other hand, using the parameter set P_2 , the GAPs obtained were less than 4%. Besides, the times of the matheuristic with the parameter set P_2 are better than the parameter set P_1 . Notice that the matheuristic using the parameter set P_1 was replicated once while using the parameter set P_2 ; it was replicated ten times. In the following Section 6, we discussed the results obtained.

6. Discussions

This section discusses the results obtained from the different methods implemented. We have proposed an MCLPAMU formulation described in Section 3.1, where we have considered mobile units. Comparing the MCLPAMU versus MCLPA formulations, the results show that the MCLPAMU increased at least 2.17% until 50% in the service network per each instance. Fig. 3 compares the MCLPA and MCLPAMU using the coverage percent obtained in each instance. We observed that the MCLPAMU could not solve instances 35–37 since these are the most complicated because they included many municipalities and hospitals.

We have used the municipality of Oaxaca to visually illustrate the benefits of MCLPAMU. Oaxaca is the biggest state in Mexico. Fig. 4 shows the result obtained using the MCLPA formulation. The green zones are the municipalities covered by the service area of the hospitals, and they are marked with a red cross on the map. The blue zones

are the municipalities with access to the hospital near them. The red zones are the municipalities unattended. Notice that in Oaxaca, disjoint municipalities exist; thus, there are green zones after the blue ones.

Fig. 5 shows the result obtained using the MCLPAMU formulation; likewise, in Fig. 4, the zones in color green are the municipalities covered by the service area of the hospital or mobile unit. In this case, we have used a car with a red cross to illustrate the location of mobile units. The zones in color blue are the municipalities with access to a hospital or mobile unit, or both. The zones in red are the municipalities unattended.

Figs. 4 and 5 show the visual advantage of using MCLPAMU versus MCLPA. In the case of Oaxaca state, the service network percentage was 55.09% using MCLPA; meanwhile, the service network percentage using MCLPAMU was 96.17%. Therefore, using mobile units, the SN increased 41.08%; this behavior was similar in the rest of the instances.

Fig. 6 shows the coverage percentage between MCLPA and MCLPAMU formulations. We observed that the MCLPAMU had a more covered percentage in all instances than the MCLPA. Therefore, the use of mobile units greatly impacted the coverage. Besides, Fig. 7 shows the accessibility percentage obtained for the two formulations mentioned before. Notice that in some instances, i.e., Instance 14, the accessibility percentage was less in MCLPAMU formulation than in MCLPA. It is because the many municipalities that had already been covered then did not need to get access.

Table 5 shows the results obtained with the MCLPA formulation versus the matheuristic using the parameter P_1 and the parameter set P_2 . Notice that the matheuristic can solve more large instances than the MCLPAMU formulation. Besides, the average GAPs obtained for the matheuristic with parameter sets P_1 and P_2 are less than 0.26%

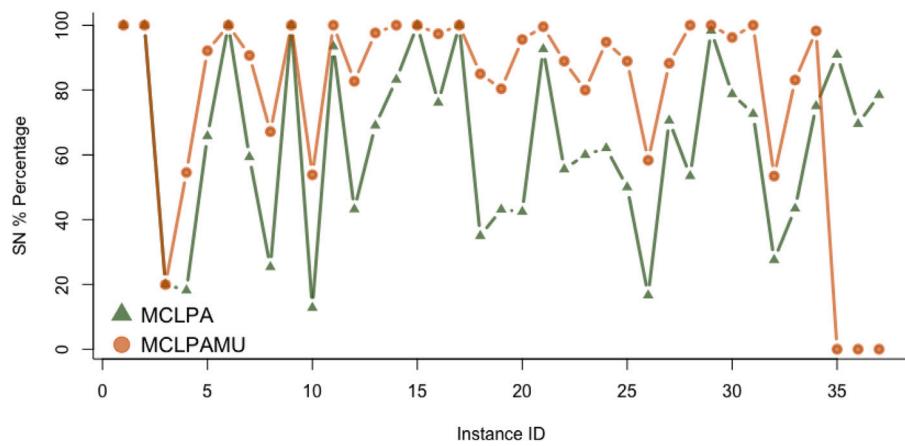


Fig. 3. Service Network (%) between MCLPA versus MCLPAMU.

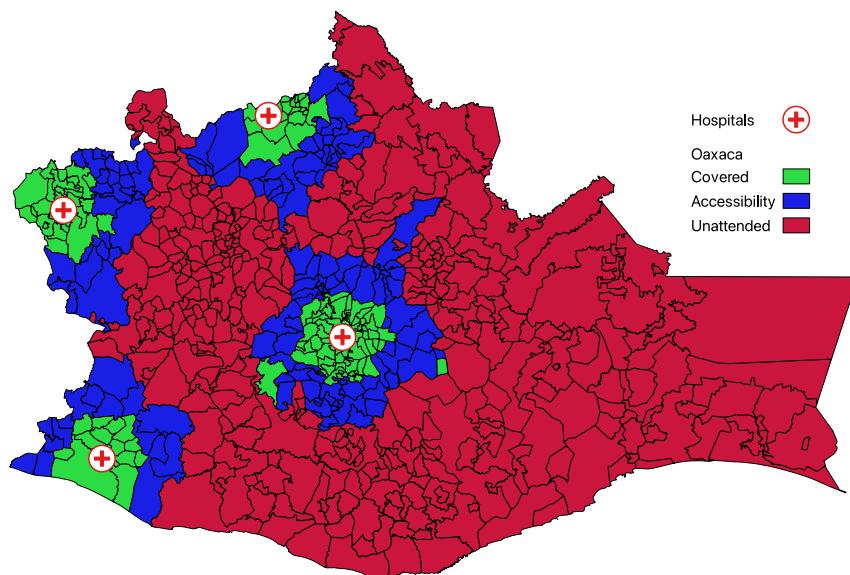


Fig. 4. Visual result of MCLPA formulation in Oaxaca state.

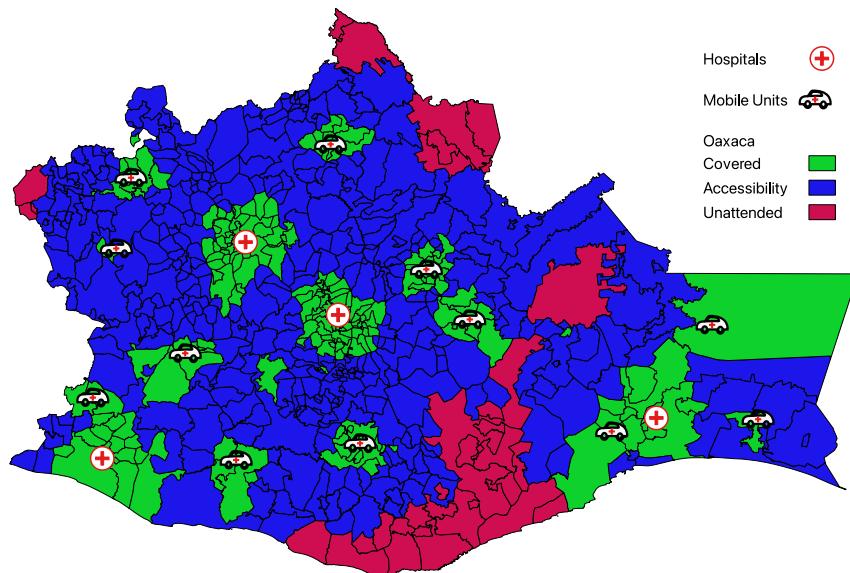


Fig. 5. Visual result of MCLPAMU formulation in Oaxaca state.

Table 5
Results of the MCLPAMU formulation and the matheuristic.

Instance	MCLPAMU		Matheuristic (P_1)			Matheuristic (P_2)			
	SN (%)	Time (s)	SN (%)	Time (s)	GAP (%)	Avg SN (%)	Avg time (s)	Avg GAP (%)	Best SN (%)
1	100.00	0.03	100.00	153.95	0.00	100.00	6.90	0.00	100.00
2	100.00	0.02	100.00	44.04	0.00	100.00	6.11	0.00	100.00
3	20.00	0.01	20.00	97.78	0.00	20.00	4.72	0.00	20.00
4	54.55	0.02	54.55	159.55	0.00	54.55	6.17	0.00	54.55
5	92.11	0.16	92.11	170.17	0.03	92.11	24.76	0.01	92.11
6	100.00	0.06	100.00	146.06	0.00	100.00	6.64	0.00	100.00
7	95.76	3.20	95.76	1680.42	0.00	95.68	177.96	0.07	95.76
8	67.16	0.31	67.16	405.31	0.02	67.16	57.85	0.00	67.16
9	100.00	0.08	100.00	25.21	0.00	100.00	7.84	0.00	100.00
10	53.85	0.06	53.85	842.70	0.00	53.85	44.27	0.00	53.85
11	100.00	3.87	100.00	210.85	0.02	100.00	38.91	0.01	100.00
12	87.65	0.64	87.65	849.73	0.00	87.28	132.37	0.29	87.65
13	100.00	43.16	100.00	1314.44	0.00	100.00	82.06	0.02	100.00
14	100.00	9.08	100.00	1612.72	0.05	100.00	351.32	0.02	100.00
15	100.00	170.22	100.00	588.14	0.00	100.00	170.02	0.00	100.00
16	97.35	11.79	97.35	1161.77	0.02	97.35	169.95	0.01	97.35
17	100.00	0.30	100.00	292.51	0.00	100.00	16.39	0.00	100.00
18	85.00	0.04	85.00	245.47	0.00	85.00	11.37	0.00	85.00
19	80.39	0.10	80.39	940.59	0.00	80.39	47.83	0.00	80.39
20	96.67	17010.10	96.67	84386.20	0.00	96.05	13048.98	1.03	96.67
21	100.00	30.51	100.00	4431.21	0.00	100.00	643.45	0.00	100.00
22	88.89	0.04	88.89	227.24	0.00	88.89	9.59	0.00	88.89
23	80.00	0.04	80.00	98.57	0.00	80.00	7.62	0.00	80.00
24	94.83	0.54	94.83	366.16	0.00	94.83	44.28	0.01	94.83
25	100.00	0.13	100.00	120.44	0.00	100.00	16.01	0.42	100.00
26	61.11	0.47	61.11	536.50	0.00	60.56	80.96	0.68	61.11
27	94.12	0.10	94.12	147.20	0.00	94.12	8.00	0.00	94.12
28	100.00	0.32	100.00	183.66	0.03	99.77	31.25	0.29	100.00
29	100.00	0.11	100.00	358.05	0.00	100.00	14.85	0.00	100.00
30	97.17	19.06	97.17	3386.02	0.01	97.17	680.34	0.01	97.17
31	100.00	5.30	100.00	3085.71	0.00	100.00	194.29	0.00	100.00
32	53.45	0.72	53.45	468.77	0.00	53.45	21.70	0.00	53.45
33	83.84	324.80	72.98	19609.80	7.40	78.89	11639.42	3.17	79.80
34	98.25	259.48	97.01	21591.50	1.41	98.13	9088.57	0.12	98.25
35	-	-	98.86	33915.20	-	98.86	21822.88	-	98.86
36	-	-	96.65	142231.00	-	97.62	59087.68	-	97.99
37	-	-	-	-	-	-	-	-	-

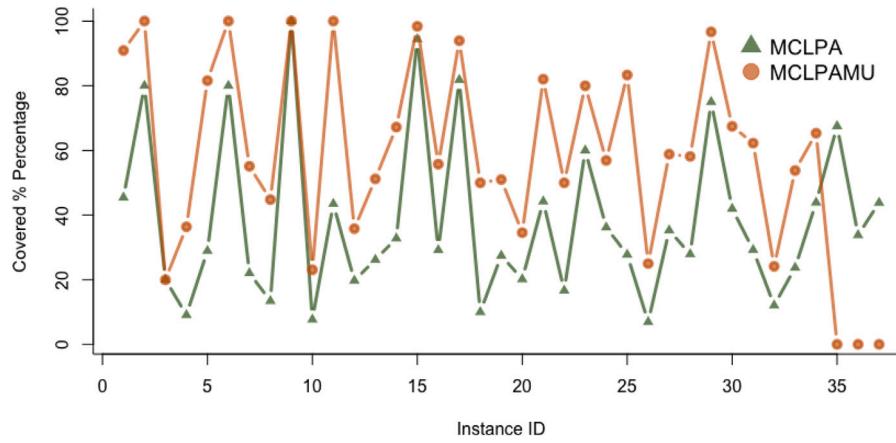


Fig. 6. Coverage (%) between MCLPA versus MCLPAMU.

and 0.18%, respectively. However, the time increased exponentially using the parameter P_1 since the matheuristic solves each feasible solution the EDA gave. In addition, the population size was big (883); hence the time was considerable. On the other hand, we used the parameter set P_2 , and the population size was 48. Besides, we have used the repair method mentioned before. Therefore, the matheuristic has a better behavior (time and results) using the parameter set P_2 . Fig. 8 shows the computational times between the matheuristic using the parameter sets P_1 (line green) and P_2 (line orange), respectively. In general, the time of the matheuristic using the parameter set P_2 is

better than the parameter set P_1 , and the average time improvement is of 63.87%. Notice that we have used a logarithmic scale.

7. Conclusions and future work

We studied the Maximal Covering Location Problem with Accessibility Indicators and Mobil Units (MCLPAMU). The main advantage of this problem is that open facilities may deploy mobile units to increase the coverage, accessibility, distance, and opportunities of the

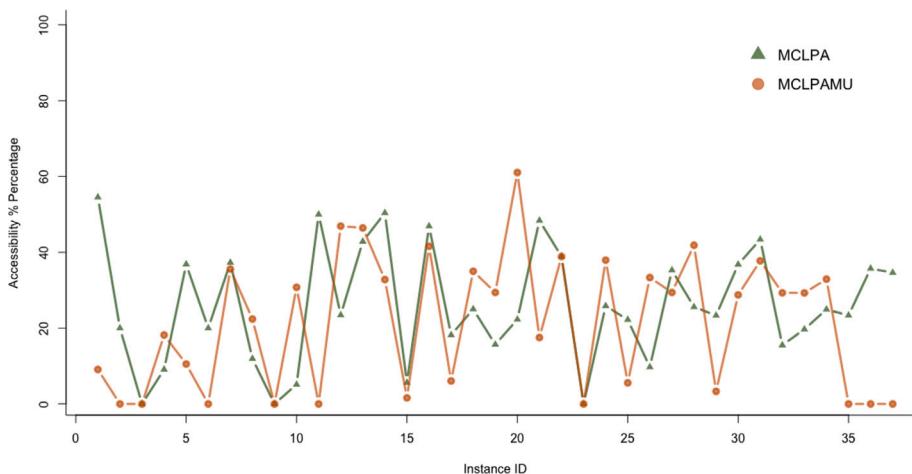
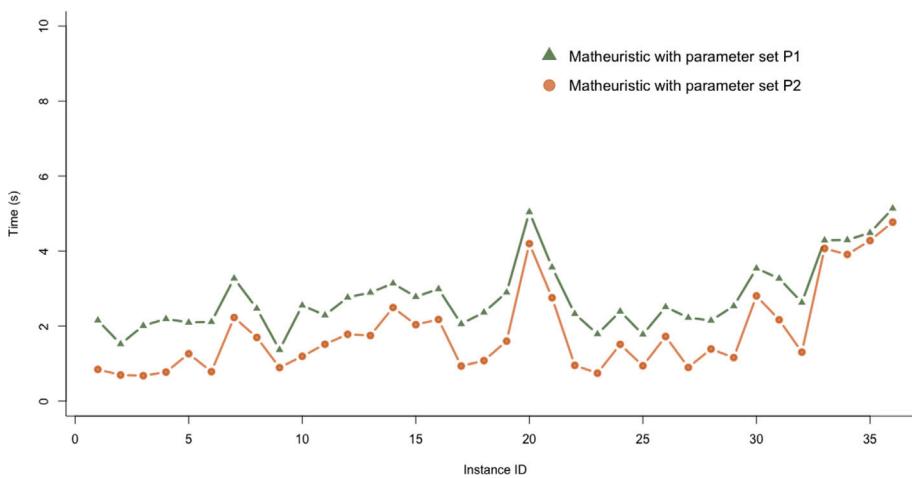


Fig. 7. Accessibility (%) between MCLPA versus MCLPAMU.

Fig. 8. Computational time in seconds of the matheuristic between parameter sets P_1 and P_2 .

demand zones. We formulated the MCLPAMU as a mixed-integer linear programming model that solves small and medium instances. For larger instances, we proposed a matheuristic based on an Estimation of Distribution Algorithm (EDA) metaheuristic and a parameterized MCLPAMU formulation. The EDA showed good performance and robustness for these types of problems. On the other hand, we have used measures such as poverty index, population number, hospital locations, and COVID-19 patients. They are essential for opening hospitals and deploying mobile units near the municipalities. The information used in this work was obtained from public and available databases for anyone interested in them. Besides, we created a database with the coordinates of COVID-19 hospitals in Mexico, also available on the web. The experimental results show that the MCLPAMU exact formulation can obtain quality results for small and medium instances in less than one hour. On the other hand, the MCLPAMU obtained an average improvement of 21.67% versus the MCLPA. For large instances, our matheuristic gets quality results and an average improvement of 21.15% versus the MCLPA. The GAPs obtained in small and medium instances in the matheuristic are less than 4% using a repair method and the parameter set P_2 while using the parameter set P_1 without repair method, the GAPs are less than 8%. On the other hand, the matheuristic average time between parameter set P_1 and P_2 is 9057.91 and 3272.31 seconds, respectively. Therefore, the best parameter set is P_2 since it obtains the best GAPs and times. Comparing the MCLPAMU versus the MCLPA, the average improvement for the small instances is 21.59% while that for medium instances, the average improvement is 23.03%. In the same

way, we are comparing the matheuristic with parameter set P_2 versus the MCLPA, the average improvement for the small instances is 20.91% and for medium instances, the average improvement is 22.69%. This little difference is because Mexico is divided by economic zone for the medium instances; then, the states can share mobile units between them. This characteristic is important because cooperative coverage is possible between states.

In future work terms, it is an important study area to add capacity and demand constraints in facilities and mobile units. Even though we know that adding these restrictions to the problem will be harder to solve than the actual problem, it will be more interesting to analyze and study. Alternatively, another proposal that should be interesting to look at is a Constraint Programming (CP) model. Recently the CP has become relevant due to its actual improvements. On the other hand, another opportunity area is the study of matheuristics that consider dependencies between variables to find statistical information about the relation between facilities with mobile units and demand zones. This statistical information might be found by implementing metaheuristics that consider dependency variables like the Bivariate Marginal of Distribution Algorithm (BMDA) or EDA-Tree, the last based on dependency trees. Although, these metaheuristics are more expensive in the computational time since it considers all these dependencies. Alternatively, the dependencies can be studied a priori using learning techniques, i.e., machine learning and deep learning, among others.; incorporating learning techniques in combinatorial optimization problems is an important study area with many opportunities.

CRediT authorship contribution statement

Salvador J. Vicencio-Medina: Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Visualization. **Yasmin A. Rios-Solis:** Conceptualization, Methodology, Validation, Investigation, Writing – original draft, Supervision. **Omar Jorge Ibarra-Rojas:** Methodology, Resources, Investigation, Writing – review & editing. **Nestor M. Cid-Garcia:** Writing – original draft, Visualization, Supervision. **Leonardo Rios-Solis:** Conceptualization, Writing – review & editing, Funding acquisition.

Data availability

I have shared the link to my data (<https://doi.org/10.6084/m9.figshare.20460867.v3>).

Acknowledgments

Salvador J. Vicencio-Medina wishes to acknowledge graduate scholarship from the Mexican Science and Technology Council (CONACYT) grant 777260, and the University of Edinburgh, UK, through the 20042 SFC-GCRF Covid-19 Fund. Omar Jorge Ibarra-Rojas conducted part of this research during an academic stay at Tecnológico de Monterrey.

References

- [1] Church R, ReVelle C. The maximal covering location problem. In: Papers of the regional science association. vol. 32, (1):Springer-Verlag; 1974, p. 101–18.
- [2] Cornuejols G, Fisher M, Nemhauser GL. On the uncapacitated location problem. In: Annals of discrete mathematics. vol. 1, Elsevier; 1977, p. 163–77.
- [3] Cornuejols G, Nemhauser G, Wolsey L. The uncapacitated facility location problem. Technical Report, Cornell University Operations Research and Industrial Engineering; 1983.
- [4] Medrano-Gómez XD, Ferreira D, Toso EA, Ibarra-Rojas OJ. Using the maximal covering location problem to design a sustainable recycling network. *J. Clean. Prod.* 2020;275:124020. <http://dx.doi.org/10.1016/j.jclepro.2020.124020>, <https://www.sciencedirect.com/science/article/pii/S0959652620340658>.
- [5] Mrkela L, Stanimirović Z. A variable neighborhood search for the budget-constrained maximal covering location problem with customer preference ordering. *Oper Res* 2021;1–12.
- [6] Yin P, Mu L. Modular capacitated maximal covering location problem for the optimal siting of emergency vehicles. *Appl. Geogr.* 2012;34:247–54. <http://dx.doi.org/10.1016/j.apgeog.2011.11.013>, <https://www.sciencedirect.com/science/article/pii/S0143622811002360>.
- [7] Olgac T, Toz AC. Determining the optimum location of ground control stations (GCSs) for unmanned aerial vehicles (UAVs) in marine search and rescue (MSAR) operations. *Int J Aeronaut Space Sci* 2022;1–12.
- [8] Mendoza-Gómez R, Ríos-Mercado RZ. Location of primary health care centers for demand coverage of complementary services. *Comput. Indust. Eng.* 2022;169:108237. <http://dx.doi.org/10.1016/j.cie.2022.108237>, <https://www.sciencedirect.com/science/article/pii/S0360835222003072>.
- [9] Ibarra-Rojas O, Ozuna L, López-Piñón D. The maximal covering location problem with accessibility indicators. *Socio-Econ Plan Sci* 2020;71:100758.
- [10] World Health Organization. WHO Coronavirus (COVID-19) Dashboard. 2022, URL <https://covid19.who.int>.
- [11] Guaman-Bautista LP, Moreta-Urbano E, Oña-Arias CG, Torres-Arias M, Kyriakis NC, Malci K, et al. Tracking SARS-CoV-2: Novel trends and diagnostic strategies. *Diagnostics* 2021;11(11):1981.
- [12] Malci K, Walls LE, Rios-Solis L. Rational design of CRISPR/Cas12a-RPA based one-pot COVID-19 detection with design of experiments. *ACS Synth Biol* 2022;11(4):1555–67.
- [13] Boonmee C, Arimura M, Asada T. Facility location optimization model for emergency humanitarian logistics. *Int J Disaster Risk Reduct* 2017;24:485–98.
- [14] Mousavi H, Darestani SA, Azimi P. An artificial neural network based mathematical model for a stochastic health care facility location problem. *Health Care Manag Sci* 2021;24(3):499–514.
- [15] Grot M, Becker T, Steenweg PM, Werners B. Enhanced coverage by integrating site interdependencies in capacitated EMS location models. *Health Care Manag Sci* 2021. <http://dx.doi.org/10.1007/s10729-021-09562-4>, Advance online publication.
- [16] Ahmadi-Javid A, Seyed P, Syam SS. A survey of healthcare facility location. *Comput Oper Res* 2017;79:223–63. <http://dx.doi.org/10.1016/j.cor.2016.05.018>, URL <https://www.sciencedirect.com/science/article/pii/S0305054816301253>.
- [17] Stummer C, Doerner K, Focke A, Heidenberger K. Determining location and size of medical departments in a hospital network: A multiobjective decision support approach. *Health Care Manag Sci* 2004;7(1):63–71.
- [18] Doerner K, Focke A, Gutjahr WJ. Multicriteria tour planning for mobile healthcare facilities in a developing country. *European J Oper Res* 2007;179(3):1078–96.
- [19] Gendreau M, Laporte G, Semet F. A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel Comput* 2001;27(12):1641–53.
- [20] Doerner K, Gutjahr W, Hartl R, Karall M, Reimann M. Heuristic solution of an extended double-coverage ambulance location problem for Austria. *CEJOR Cent Eur J Oper Res* 2005;13(4).
- [21] Erdemir ET, Batta R, Rogerson PA, Blatt A, Flanigan M. Joint ground and air emergency medical services coverage models: A greedy heuristic solution approach. *European J Oper Res* 2010;207(2):736–49.
- [22] Curtin KM, Hayslett-McCall K, Qiu F. Determining optimal police patrol areas with maximal covering and backup covering location models. *Netw Spat Econ* 2010;10:125–45.
- [23] Abounacer R, Rekik M, Renaud J. An exact solution approach for multi-objective location–transportation problem for disaster response. *Comput Oper Res* 2014;41:83–93.
- [24] Jenkins PR, Lunday BJ, Robbins MJ. Robust, multi-objective optimization for the military medical evacuation location-allocation problem. *Omega* 2020;97:102088.
- [25] Azizan M, Go T, Hatta W, Lim C, Teoh S. Comparison of emergency medical services delivery performance using maximal covering location and gradual cover location problems. *Int J Electr Comput Eng.* 2017;7(5):2791–7.
- [26] Berman O, Drezner Z, Krass D, Wesolowsky GO. The variable radius covering problem. *European J Oper Res* 2009;196(2):516–25.
- [27] Forghani Youshanlo M, Sahraeian R. Dynamic multi-objective maximal covering location problem with gradual coverage. In: Cortés P, Maeso-González E, Escudero-Santana A, editors. Enhancing synergies in a collaborative environment. Cham: Springer International Publishing; 2015, p. 39–47.
- [28] Karataş M, Razib N, Tozanc H. A comparison of p-median and maximal coverage location models with Q-coverage requirement. *Procedia Eng* 2016;149:169–76.
- [29] Taiwo OJ. Maximal covering location problem (mcpl) for the identification of potential optimal covid-19 testing facility sites in nigeria. *Afr. Geogr. Rev.* 2021;40(4):395–411. <http://dx.doi.org/10.1080/19376812.2020.1838306>.
- [30] Hassan S, Alnowibet K, Agrawal P, Ali Wagdy M. Optimum location of field hospitals for COVID-19: A nonlinear binary metaheuristic algorithm. *Comput Mater Continua* 2021;68(1):1183–202.
- [31] Calderon S, Nicolis O, Peralta B, Menares F. Predicción de casos COVID-19 y modelo de localización asignación de bases y ambulancias considerando factores de vulnerabilidad. *Ingen Revis Chil de Ingen* 2021;29(3):564–82.
- [32] Kuvvetli Y. A goal programming model for two-stage covid19 test sampling centers location-allocation problem. *Cent. Eur. J. Oper. Res.* 2023;31(1):1–20.
- [33] Lusiantoro L, Mara S, Rifai A. A locational analysis model of the COVID-19 vaccine distribution. *Oper Supply Chain Manag: Int J* 2022;15(2):240–50. <http://dx.doi.org/10.31387/oscm0490344>.
- [34] Makr M-C, Folkesson C, et al. Accessibility measures for analyses of land-use and travelling with geographical information systems. In: Proceedings from the annual transport conference at aalborg university. vol. 6, (1). 1999.
- [35] Ibarra-Rojas O, Hernandez L, Ozuna L. The accessibility vehicle routing problem. *J Clean Prod* 2018;172:1514–28.
- [36] Scherer D, Moeini M, Wendt O. A matheuristic for the vehicle routing problem with drones and its variants. *Transp Res C* 2019;106:166–204.
- [37] Gunawan A, Widjaja AT, Vansteenwegen P, Vincent FY. A matheuristic algorithm for the vehicle routing problem with cross-docking. *Appl Soft Comput* 2021;103:107163.
- [38] Souto G, Morais I, Mauri GR, Ribeiro GM, González PH. A hybrid matheuristic for the Two-Stage Capacitated Facility Location problem. *Expert Syst Appl* 2021;185:115501.
- [39] Boschetti M, Maniezzo V. A set covering based matheuristic for a real-world city logistics problem. *Int Trans Oper Res* 2015;22(1):169–95.
- [40] Hauschild M, Pelikan M. An introduction and survey of estimation of distribution algorithms. *Swarm Evol Comput* 2011;1(3):111–28.
- [41] Larrañaga P, Lozano JA. Estimation of distribution algorithms: A new tool for evolutionary computation. vol. 2, Springer Science & Business Media; 2001.
- [42] Pelikan M, Hauschild MW, Lobo FG. Introduction to estimation of distribution algorithms. *MEDAL Rep* 2012;2012003.
- [43] Mühlénbein H, Paass G. From recombination of genes to the estimation of distributions I. Binary parameters. In: International conference on parallel problem solving from nature. Springer; 1996, p. 178–87.
- [44] Mühlénbein H. The equation for response to selection and its use for prediction. *Evol Comput* 1997;5(3):303–46.
- [45] Maria E, Budiman E, Taruk M, et al. Measure distance locating nearest public facilities using haversine and euclidean methods. In: Journal of physics: Conference series. vol. 1450, (1):IOP Publishing; 2020, 012080.

- [46] Cintrano C, Chicano F, Alba E. Using metaheuristics for the location of bicycle stations. *Expert Syst Appl* 2020;161:113684.
- [47] Velasco J, Vicencio S, Lozano JA, Cid-Garcia NM. Delineation of site-specific management zones using estimation of distribution algorithms. *Int Trans Oper Res* 2021.
- [48] Achamrah FE, Riane F, Di Martinely C, Limbourg S. A matheuristic for solving inventory sharing problems. *Comput Oper Res* 2022;138:105605.
- [49] López-Ibáñez M, Dubois-Lacoste J, Cáceres LP, Birattari M, Stützle T. The irace package: Iterated racing for automatic algorithm configuration. *Oper Res Perspect* 2016;3:43–58.
- [50] Rezaei J. Best-worst multi-criteria decision-making method: Some properties and a linear model. *Omega* 2016;64:126–30.

Salvador de Jesús Vicencio Medina is studying for his Ph.D. in engineering sciences from Tecnológico de Monterrey. He made a Master's degree in mathematical modeling and optimization from CIMAT, Mexico. His interests are operations research and its applications, specifically mathematical programming, metaheuristics, matheuristics, and artificial intelligence. Some application areas include optimization in health care and agriculture.

Yasmín Ríos Solís studied applied mathematics at ITAM, Mexico. She completed her master's and Ph.D. in combinatorial optimization at the Sorbonne University in France, and then a postdoc at the University of Bielefeld in Germany. She is currently part of the Tecnológico de Monterrey, School of Engineering and Sciences, Mexico. Her research is related to discrete optimization problems for which she proposes mathematical models and new optimization methods both exact and heuristic.

Omar Jorge Ibarra Rojas holds a Ph.D. in systems engineering from the Universidad Autónoma de Nuevo Leon (UANL). He made a postdoctoral stage of two years at the Bus Rapid Transit - Centre of Excellence. Since 2015, he is a professor in the School of Physics Mathematics at the UANL. His expertise is the implementation of mathematical programming, metaheuristics, and other operations research techniques to support the decision-making process mainly for public transport but including also general logistics and manufacturing planning.

Nestor Miguel Cid-García received the PhD. degree in Systems Engineering in 2016 from the Autonomous University of Nuevo Leon, Mexico. Since 2016 he has been a full researcher with the Chairs Program of the Mexican National Council of Science and Technology (CONACYT) in the National Laboratory of Geointelligence, CentroGeo, Aguascalientes, Mexico. His research interests include mathematical modeling; application and theory of operations research techniques; and the design, development, and analysis of efficient approximate and exact algorithms. Some application areas include delineation of site-specific management zones, agricultural production planning, special economic zones, and 2D packing problems.

Leonardo Ríos Solís is an associate professor at the Newcastle University, School of Natural and Environmental Sciences. The research at his lab focuses on promoting further understanding of the interphase of Biochemical Engineering and Synthetic Biology as well as finding novel ways to apply this knowledge to tackle the growing global socio-economic inequalities.