

Article

A Bi-Level Vaccination Points Location Problem That Aims at Social Distancing and Equity for the Inhabitants

Edith Salinas ^{1,†}, José-Fernando Camacho-Vallejo ^{2,*}  and Samuel Nucamendi-Guillén ^{3,†} 

¹ Facultad de Ciencias Físico-Matemáticas, Universidad Autónoma de Nuevo León, San Nicolás de los Garza 66450, Nuevo León, Mexico; edith.salinasd@uanl.edu.mx

² Tecnológico de Monterrey, Escuela de Ingeniería y Ciencias, Ave. Eugenio Garza Sada 2501, Monterrey 64849, Nuevo León, Mexico

³ Facultad de Ingeniería, Universidad Panamericana, Álvaro del Portillo 49, Zapopan 45010, Jalisco, Mexico; snucamendi@up.edu.mx

* Correspondence: fernando.camacho@tec.mx

† These authors contributed equally to this work.

Abstract: Designing efficient vaccination programs that consider the needs of the population is very relevant to prevent reoccurrence of the COVID-19 pandemic. The government needs to provide vaccination points to give out vaccine doses to the population. In this paper, the authors analyze the location of vaccination points whilst addressing the inhabitants' preferences. Two objectives that prevent crowding of inhabitants are considered. The government aims for the minimum distance between located vaccination points is maximized, and for the number of inhabitants that attend the different vaccination points to be equitable. One of the key aspects of this problem is the assumption that inhabitants freely choose the located vaccination point to go. That decision affects the objectives of the government, since crowding at vaccination points may appear due to the inhabitants' decisions. This problem is modeled as a bi-objective, bi-level program, in which the upper level is associated to the government and the lower level to the inhabitants. To approximate the Pareto front of this problem, a cross-entropy metaheuristic is proposed. The algorithm incorporates criteria to handle two objective functions in a simultaneous manner, and optimally solve the lower-level problem for each government decision. The proposed algorithm is tested over an adapted set of benchmark instances and pertinent analysis of the results is included. An important managerial insight is that locating far vaccination points does not lead us to a more equitable allocation of inhabitants.

Keywords: bi-level programming; COVID-19; public facility location; vaccination system; cross-entropy metaheuristic



Citation: Salinas, E.; Camacho-Vallejo, J.F.;

Nucamendi-Guillén, S. A Bi-Level Vaccination Points Location Problem That Aims at Social Distancing and Equity for the Inhabitants. *Axioms*

2023, 12, 305. <https://doi.org/10.3390/axioms12030305>

Academic Editor: Hsien-Chung Wu

Received: 31 January 2023

Revised: 8 March 2023

Accepted: 15 March 2023

Published: 17 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In December 2019, the World Health Organization wrote about the existence of a new virus: SARS-CoV-2, which causes the COVID-19 disease. Some of its common symptoms are fever, dry cough, and tiredness. Measures to prevent the spread of COVID-19 include physical distancing from other people, face masks, washing one's hands periodically, and covering one's mouth with a flexed elbow or a handkerchief when coughing. As of 11 September 2022, 605 million confirmed cases and 6.4 million deaths have been reported worldwide.

However, the vaccine against this virus is already available. Vaccination of inhabitants seems to be the safest and most effective way to control the COVID-19 pandemic. In the Ref. [1], the authors state that a single dose of vaccine can be a drop in the ocean, but several doses together can save a population.

During the worst of the pandemic and up to this day, there has been high acceptance of the COVID-19 vaccine, and strong demand for it. This is the main reason for designing

appropriate vaccination programs is of utmost importance. Having appropriate vaccination programs according to the needs of the population to be vaccinated is an essential element to prevent the re-emergence of this pandemic.

Our research is oriented to humanitarian logistics, looking at cases where the government needs to locate different vaccination points for the inhabitants of a certain region, in such a way that there is the most equitable access to vaccines. Vaccination of inhabitants seems to be the safest and most effective way to control the COVID-19 pandemic. In the Ref. [1], the authors state that a single dose of vaccine can be a drop in the ocean, but several doses together can save a population.

In this paper, a problem concerning the vaccination strategic decisions is studied. The government needs to locate different vaccination points (VPs). In trying to maintain the healthy distance and avoid agglomerations, two objectives are pursued: (i) that the located VPs are as far as possible from each other, and (ii) that inhabitants are equally distributed (as much as possible) within the VPs.

The vaccination strategy imposes some mobility constraints, such as, only the current inhabitants of the municipality will receive the vaccine in the VPs of that specific municipality. However, the government cannot impose this restriction on an inhabitant, each inhabitant is free to attend the VP he/she prefers. This assumption leads us to take into account the inhabitants' preferences towards the VPs. Therefore, the allocation of inhabitants to VPs is chosen by each inhabitant and not by the government.

In summary, the government decides the location of VPs within the municipality and the inhabitants attend the VP they prefer. It is evident that the inhabitants' decisions directly affect the crowding in the VPs, which is of interest to the government. Hence, due to the existing hierarchy in this situation and the relationship between the decisions, a bi-level programming approach is suitable to study this problem. Two objectives are considered at the upper level, simultaneously, and this prevents application of the well-known single-level reformulation in an efficient manner. That is, despite the fact that the bi-level model may be reformulated as an equivalent single-level one by using the optimality conditions of the lower level, the analysis must not lose sight of the fact that two objective functions are being optimized at the lower level. Thus, a bi-objective single-level complex model must be solved.

In order to handle the inherent complexity of this bi-objective bi-level problem, a cross-entropy algorithm is proposed to approximate the Pareto front of this VPs location problem. Several adaptations are needed to design the proposed algorithm. Firstly, an adaptation is made to guide the search while considering the two objective functions. Secondly, a strategy to deal with the parameterized lower-level problem at each stage of the algorithm is designed.

We can highlight five main contributions of this research to the literature. First, the combination of a p -dispersion problem with a facility location problem that balances the allocation of the users in the located facilities is performed for the first time. This can be considered as an extension of the p -dispersion problem. Second, a bi-objective bi-level program is proposed to model the problem under study. Third, the proposed model is applied to a vaccination system that aims to maintain the healthy distance among inhabitants without imposing restrictions regarding the VPs that they need to attend. In the literature, there are no papers with those characteristics, that is, neither the proposed variant of the p -dispersion problem nor the application to the location of VPs. Fourth, the proposal of a cross-entropy metaheuristic that simultaneously deals with two objectives, and handles the bi-level structure of the problem. That is, it solves the lower-level problem while considering the upper-level decisions. This is the first time that cross-entropy is applied towards solving a bi-objective bi-level problem, so the specific components for dealing with both objectives and the lower-level are carefully designed. Fifth, extensive computational experimentation is conducted to validate the proposed metaheuristic and to obtain important managerial insights regarding the location of VPs and social distancing.

The rest of this paper is organized as follows. Section 2 reviews the literature of facility location problems, users' preferences and applications related to COVID-19 pandemic. Section 3 presents the bi-level formulation of the problem. Section 4 describes the proposed metaheuristic and its components. Section 5 presents the computational results on adapted test instances and the respective analyses. Finally, Section 6 outlines the conclusions of our research and lists some interesting further research directions.

2. Literature Review

The literature is divided in three main directions. First, a general overview of facility location problems is presented. Particular attention is focus on the p -dispersion problem and the ones that balance allocation of users. Second, some previous researches that consider the preferences of the users in the allocation phase are reviewed. Third, a review on vaccine logistics, in general, with special emphasis on systems for the COVID-19 pandemic is conducted.

2.1. Facility Location

Facility location problems (FLPs) have been intensively studied during the last decades under different assumptions. In general, FLPs consist of the following: given a set of potential locations and a set of users, two decisions have to be made, (i) where to locate the facilities and (ii) which users allocate in each located facility. Decisions are taken in such a way that a predetermined objective function is optimized [2].

Due to the versatility of FLPs, many different variants have been proposed in the literature. For instance, the consideration of a limited capacity in the facilities, a coverage radius, users with unitary demand, balanced facilities with respect of the allocated users, the location of a predetermined number of facilities, among others. In the Refs. [3,4] some variants and applications are presented. Additionally, their corresponding mathematical models and the proposed methods for solving each specific problem are detailed.

In particular, we are interested in two closely related and well-known FLPs. The first one studies the case when p facilities must be located but aiming to be as far as possible. This problem is called as the p -dispersion problem [5,6]. Since the objective is to maximize the minimum distance between located facilities, the allocation of users is irrelevant. In many contexts it is important to bear in mind the allocation of users. That is one of the reasons why allocation is considered in the problem under study.

The second one is the FLP that balances the allocation of users in the located facilities. This problem is proposed in the Ref. [7], in which the objective is to minimize the difference between the maximum and the minimum number of users allocated to the facilities. In that problem, users are allocated to the nearest located facilities.

As mentioned above, one of the main contributions of our study was to combine these two FLPs in the same model. The nature of the original p -dispersion problem neglects the allocation of users. However, there are situations in which it is important to be considered. Additionally, the balance of users allocated to each located facility may be pursued to enhance stability of the system. In other words, it is aimed at that the workload of the facilities is similar. For the best knowledge of the authors, this is the first time that these two problems have been merged into a single one.

2.2. Allocation Based on the Users' Preferences

Preferences are an important factor that are commonly included to consider the behavior of users in allocation problems. For example, in the stable marriage problem, the elements of one set place their preferences towards the elements of the other set, and vice versa [8,9]. Another approach when considering preferences in the decision-making process corresponds to the allocation of users in FLPs. In this case, users patronize the facilities based on their own preferences, which can be based on attractiveness, proximity, and quality, among others. This may be seen as the allocation rule followed by the users in a facility location scheme.

Users' preferences were considered for the first time in FLPs in the Ref. [10], in which users ordered the potential facilities based on their own preferences. Once some facilities are located, the ordered list of preferences determines the allocation of users. Since then, different FLPs have considered the preferences of the users, such as in the Refs. [11,12]. Moreover, considering users' preferences yields to a hierarchized structure of the problem, which can be naturally modeled with bi-level programming [13–16].

Recently, a bi-level programming problem that balances the number of allocated users to the located facilities while considering users' preferences was studied in the Ref. [17]. The main difference of our study with respect to the Ref. [17] is that we are considering two objective functions simultaneously. Therefore, the algorithm designed in that research cannot be easily extended to our bi-objective problem. Considering two objective functions complicates the guidance towards improving solutions. That model arises from a 5G network's configuration context.

2.3. Applications Related to COVID-19

It is evident that the emergence of COVID-19 has generated profound social and economic changes. Consequently, different approaches must be enabled to prevent and control this disease. For this reason, in the last couple of years, significant efforts have been dedicated to the COVID-19 pandemic. For example, studies related to the supply chain [18–20], allocation of patients [21–23], distribution of test kits [24,25], medicines and medical equipment distribution [26,27], and location of COVID-19 testing centers [28–30].

Since the start of the pandemic, we have been facing an unprecedented race to develop a vaccine. Successful efforts have been identified and vaccines are available. However, the production rate is not enough to vaccinate the entire global population. Therefore, the limited amount of available vaccines creates another great challenge: how can we distribute vaccines to mitigate the impact of the virus near the end of the pandemic?

In recent years, there has been an increasing interest in vaccination logistics [31]. For example, studies that consider equitable distribution of the vaccines can be found in the Refs. [32–35]. Particularly, in the Ref. [36], a model that seeks to optimize the distribution strategy of vaccines is proposed. The main decision of the problem is to locate vaccination centers within a nation. They conclude that determining the number and location of vaccination centers is also essential to ensure that vaccines are available to the entire population. There are some studies where it is studied how to locate and distribute vaccines, for example in the Ref. [35], and the authors present a mixed-integer linear programming model for inventory and location problems. However, they focus on an equitable distribution of influenza vaccines in developing countries during the pandemic. Their proposed model uses an objective function that aims to distribute vaccines in an equitable manner. They present a case study from a developing country to showcase the effectiveness and demonstrate the applicability of their proposed optimization model.

It is worth mentioning that due to the rapid spread of COVID-19, the demand for emergency medical facilities has grown tremendously. Regarding this topic, in the Ref. [37] the authors addressed a situation in which location and allocation problems were integrated, aiming to help in managing the main public health emergencies. It aims at minimization of the time spent in the locations and the assignment of emergency medical facilities in case of the occurrence of a public health emergency. In addition, a genetic algorithm is proposed to provide solutions to the problem.

The allocation of vaccines while considering factors such as geographical region, operational limitations, capacity of medical centers, and availability of vaccines are considered in the Refs. [38,39]. In the first one, that is, in the Ref. [38], a decision process in which vaccinations are carried out in two phases to contain the outbreak of an infectious disease in a set of geographical regions was considered. In the first phase, a limited number of vaccine doses are assigned to each region; whereas in the second phase, additional doses can be assigned to regions where the epidemic has not been contained. They formulated the vaccine allocation problem as a two-step stochastic linear program and reduced it to a

linear program for obtaining acceptable solutions. In the second one, the authors proposed a mixed-integer linear programming problem aiming to develop an effective vaccine allocation model based on the parameters of the total population susceptible to COVID-19 and the risk of exposure (see the Ref. [39]). They analyzed a case study in which the proposed model was applied to test different vaccine allocation and distribution scenarios.

An important aspect to consider with the location and allocation of vaccines is the equity. In the Ref. [32], the problem to be resolved was the minimization of the number of vaccine doses distributed to effectively extinguish an emerging outbreak in its early stages. The authors proposed an equity constraint to help health public authorities to distribute vaccines in a fairer manner when making decisions about vaccine distribution.

An important approach in the context of the pandemic is to consider multiple objectives, simultaneously, for the same problem. The latter helps to provide more managerial insights related to the location and allocation of vaccines. The reason is that a compromise between two different objective functions can be identified to evaluate different decision-making scenarios. In the Ref. [40], a multi-period vaccination planning problem is addressed. That problem minimizes the total travel distance of the inhabitants vaccinated and minimizes the operating cost of the health program. An optimal plan determines, for each period, which vaccination points to open, how many vaccination stations to enable at each site, how to assign inhabitants from different regions to open vaccination points, and the amount of restocking at each vaccination point. They formulate the problem as a mixed-integer linear bi-objective program. Initially, they propose two solution methods: the weighted sum and ϵ -constraint methods. In addition, they develop a genetic algorithm where an allocation strategy and a new dynamic programming method are included in the algorithm to obtain good feasible solutions. Additionally, a case study is presented indicating that the methods reduce operating cost and total travel distance. In the Ref. [41], a multi-target mixed-integer non-linear programming model is proposed to help the centers for disease control and prevention determine the locations of vaccination stations and, at the same time, consider travel distance, operating cost and working hours.

Another related bi-objective study appears in the Ref. [34], which seeks to optimize the design of a vaccine distribution network through a mixed-integer non-linear programming model, but while considering two objectives: the minimization of the expected total number of deaths in the population and the minimization of the total distribution costs associated with the vaccination campaign. The proposed model is linearized and validated through a real case study of the vaccination campaign against COVID-19 in France. In addition, the authors show that a vaccination strategy that combines population prioritization and quarantine restrictions leads to an 8.5% decrease in the total number of deaths.

Starting from the previous idea of prioritizing the population in the process of vaccination against COVID-19, we consider that the inhabitants are free to decide which VP to go to and be vaccinated at. Therefore, we consider the preferences of the inhabitants with respect to each of the possible VPs. It is worth highlighting that the preferences of the inhabitants that will receive a vaccination dose against COVID-19 have not been considered before in the literature. Moreover, in this problem, the government authorities do not impose on the inhabitants which VP to go to, but the VP's location clearly affects the inhabitants' decisions. This kind of hierarchy in the decision-making process has also not been considered before in the literature of a COVID-19 vaccination planning problem.

To summarize, in this study, a bi-objective bi-level programming problem is proposed. The decision-maker associated to the upper level aims to balance the inhabitants that attend each vaccination point, and also aims to locate the vaccination points the farthest from each other to motivate inhabitants to go and receive a vaccine dose. On the other hand, the decision-maker associated to the lower level aims to maximize the preferences of the inhabitants regarding the vaccination points they will approach.

2.4. Summary of the Literature Review

To better illustrate the gap in the literature that our research is filling up, the main characteristics of the closely related and above referred papers are summarized in Table 1.

Table 1. Summary of the characteristics considered in the closely related FLPs.

Reference	Objective	Biobjective	Bilevel	Preferences	Metaheuristic	COVID-19
[5]	Max min distance					
[6]	Max min distance					
[7]	Min disbalance					
[10]	Min costs			X		
[11]	Min costs			X		
[12]	Min costs		X	X		
[13]	Min costs		X	X	PSO, SA, VNS	
[14]	Min costs		X	X	EA + PR	
[15]	Min costs		X	X	EA	
[16]	Min costs		X	X	EA	
[17]	Min max disbalance		X	X	EA	
[37]	Min relief time				GA	X
[32]	Min vaccines used					X
[40]	Min costs Min travel distance	X			NSGA-II	X
[34]	Min costs Min deaths	X				X
This paper	Max min distance Min disbalance	X	X	X	BONXEA	X

It can be noticed that the characteristics of the problem under study in this research have not been considered in previous research, simultaneously. This fact highlights one of our main contributions.

3. Problem Statement and Its Mathematical Model

Consider the situation described in Section 1, in which the government needs to locate a predefined number of VPs for vaccination inhabitants against COVID-19. The inhabitants are free to attend the VP they prefer. The latter decision directly affects the crowding in the VPs, which is undesirable. This situation can be modeled under a bi-objective bi-level programming approach. The upper level is associated with the government, and the lower level to the inhabitants.

First, the sets, parameters, decision and auxiliary variables involved in the model are defined. Let I be the set of VPs that must be located, and let J be the set of all inhabitants who need to receive the vaccine. The predefined number of VPs that must be located is denoted by p . The distance between a VP $i \in I$ and the inhabitant $j \in J$ is represented by d_{ij} . Additionally, g_{ij} represents the preference that an inhabitant $j \in J$ has for a VP $i \in I$.

The binary decision variables associated with the upper level are y_i , which is 1 when VP is located in location $i \in I$, and 0 if not. On the other hand, binary decision variables of the lower level are denoted by x_{ij} , where 1 is when the inhabitant $j \in J$ receives the vaccine from VP $i \in I$, and 0 in other case.

Auxiliary continuous variables U and L compute the maximum and minimum number of inhabitants assigned to any VP, respectively.

As mentioned previously, the government aims to maintain the *healthy distance*, and prevent crowds. Therefore, two objectives are considered in this problem. The first one is to balance the number of inhabitants assigned to each VP, that is, that population is equally distributed (as possible) into the VPs. This objective function is formulated as follows:

$$\min U - L \tag{1}$$

The second objective function aims to have the VPs as far as possible from each other, that is, to maximize the minimum distance between the located VPs. This objective function is represented as follows:

$$\max \min_{i,k \in I; i < k} \{d_{ik}y_i y_k\} \tag{2}$$

The latter equation can be linearized by introducing an auxiliary continuous variable r , which indicates the minimum distance between each pair of located vaccination points. Hence, the linearized second objective function is:

$$\max r \tag{3}$$

An additional set of constraints must be included to achieve linearization of Equation (2). Let M be a sufficiently large positive constant, usually referred to as the big- M . The distance from each pair of located VPs is computed, and r will take the minimum value between all of them. This is as follows:

$$r \leq M(2 - y_i - y_k) + d_{ik} \quad \forall i, k \in I, i < k \tag{4}$$

The manner in which the linearization works is explained next. Equation (4) establishes upper bounds for the value of r . Since M is a large constant, note that this constraint is relaxed when neither the i -th nor k -th VPs are located. When two specific VPs are located, then the first term of the right-hand side is zero. In this case, r is bounded by the distance between these two VPs. Due to the sign of the inequality given in Equation (4) and by the orientation of objective function defined in Equation (3), the smaller value of the distances involved in the bounding constraints is assigned to r (which is the maximum possible one).

Consider that there is a predefined number of VPs to be located. This is ensured by the following constraint:

$$\sum_{i \in I} y_i = p \tag{5}$$

As indicated above, the upper-level decision variables are binary:

$$y_i \in \{0, 1\} \quad \forall i \in I \tag{6}$$

Once the government has chosen the VPs to be located (x), the inhabitants need to be allocated to the VPs to get vaccinated (y). Since we are assuming that the allocation of inhabitants to VPs is performed by each inhabitant and not imposed by the government, their preferences g_{ij} towards the VPs are considered. Preferences are given as a ranking of VPs, in which 1 indicates the least preferred VP and $|I|$ the most preferred one. Therefore, the maximization of these preferences is aimed at. The objective function associated with the lower level is as follows:

$$\max \sum_{i \in I} \sum_{j \in J} x_{ij} g_{ij} \tag{7}$$

There are some constraints associated with the lower level. The first one indicates that inhabitants can be allocated only to located VPs.

$$x_{ij} \leq y_i \quad \forall i \in I, \forall j \in J \tag{8}$$

Another imposed constraint must guarantee that inhabitants each receive only one vaccine. That is, a unique allocation from inhabitants to VPs is allowed.

$$\sum_{i \in I} x_{ij} = 1 \quad \forall j \in J \tag{9}$$

Let n be the total number of inhabitants that will receive a vaccine, that is, $n = |J|$. Therefore, the maximum and minimum numbers of inhabitants allocated to the located VPs are computed by the following constraints, respectively:

$$U \geq \sum_{j \in J} x_{ij} \quad \forall i \in I \tag{10}$$

$$L \leq \sum_{j \in J} x_{ij} + n(1 - y_i) \quad \forall i \in I \tag{11}$$

Note that Equation (10) counts the number of inhabitants allocated to the VPs. By considering the inequality of the type greater or equal than, the value of U that satisfies Equation (10) is bounded by the maximum number of allocations to a specific VP. On the other hand, Equation (11) differentiates between opened and closed VPs and helps to compute the minimum number of inhabitants allocated to a specific VP. In detail, if a VP is located at a specific potential site, then the second term of the right-hand side is zero. Due to the sign of the inequality, the value of L that satisfies Equation (11) is the minimum number of inhabitants allocated. If a VP is not located, inhabitants cannot be allocated by Equation (8). Thus, constraint (11) is relaxed since it is bounded by n .

Finally, lower-level decision variables are binary.

$$x_{i,j} \in \{0, 1\} \quad \forall i \in I, \forall j \in J \tag{12}$$

The resulting bi-objective bi-level programming model is defined by:

$$\begin{aligned} & [Equation (1), Equation (3)] \\ & s.t. : Equations (4)–(6) \end{aligned}$$

where for a fixed y , x optimally solves:

$$\begin{aligned} & Equation (7) \\ & s.t. : Equations (8)–(12) \end{aligned}$$

It is worthwhile to note that if each of the inhabitants have a different order of preferences for the located VPs, then a unique optimal solution of the lower level exists (see the Ref. [12]). However, this may be unrealistic in our problem due to the large number of inhabitants in comparison with the located VPs. Thus, it is expected that more than one inhabitant will have the same ranking of preferences, which leads to the existence of multiple optimal solutions for the lower level. In that case, different optimal lower-level solutions, for a fixed upper-level decision, could lead to different values of the upper-level objective function. The optimistic or pessimistic approaches can be assumed to handle that issue [42,43]. In the former, the lower-level decision-maker selects the optimal solution that is more convenient to the upper-level decision-maker (cooperative approach), while in the latter, the solution that most negatively affects the upper-level decision-maker is selected by the lower-level decision-maker. In this study, the optimistic approach is followed.

The latter assumption is necessary to have a well-defined bi-level programming problem. Nevertheless, in the problem under study, it is not straightforward to identify the optimistic approach since we are simultaneously optimizing two objective functions. Note that both objective functions can be seen as separable, that is, Equation (1) is determined by the lower-level variables, and Equation (3) is only defined by upper-level variables.

Therefore, the optimistic approach relies on choosing the optimal solution of the lower level that is more convenient for balancing the inhabitants that attend the vaccination points.

To obtain all the possible multiple optimal solutions of the lower level, the corresponding separation problem is identified [44]. That is, all the allocations of inhabitants to located VPs are identified and the appropriate constraint is added. This means it is prohibited that all the identified allocations are repeated in the augmented problem. Additionally, a constraint that guarantees that the lower-level objective function value is maintained is included. Then, the augmented lower-level problem is solved. If it is infeasible, it implies that no more optimal lower-level solutions exist. In the other case, the optimal solutions are recorded and at the end, the one that offers greater balance for the crowding in the VPs is selected.

4. A Bi-Objective Nested Cross-Entropy Algorithm (BONXEA)

Cross-entropy (XE) is an iterative method with two main phases. First, it creates a random sample of solutions by a specific mechanism. Then, it updates the parameters involved in the solutions generator. The latter is achieved by considering the characteristics of the previous random sample in order to create improved solutions in the next iteration.

XE has a relatively short history in optimization problems. Rubinstein developed XE as a method for estimating the probability of rare events in complex stochastic networks [45]. A few years later, XE was applied for the first time in the context of combinatorial optimization, which inspired its application in diverse applications in operational research: capacitated facility location [46–48], closed-loop supply chain in the Ref. [49], sustainable supply chain in the Ref. [50], job-shop scheduling [51], facility layout [52], and water distribution systems [53], among others.

Described in detail, XE consists of the repeated execution of the following steps [54]:

1. Generate a random sample of solutions based on a prespecified probability distribution function.
2. Use the random sample to select an elite set that is used to modify the parameters of the probability distribution function seeking to produce a sample with better objective function values in the next iteration.

The general idea is to obtain a sequence of parameter values that are being updated in an appropriate manner from one iteration to the next one. If successful, the search converges to the global optimum, or at least, to a local optimum.

XE has been successfully applied in discrete problems related to the papers studied herein. For example, in the Ref. [46] a location-allocation problem was studied. Therein, capacity and fixed costs of the facilities are considered, and the minimization of the total cost is aimed. A non-linear mixed-integer formulation is proposed and the XE method is implemented to solve the problem. Three different stages occur, each with different density functions. In the first stage, a location problem with coverage is solved by using a multivariate normal function. In the second stage, the allocation is made by following a multinomial density function. Then, in the third stage, a continuous single-facility location problem is solved. The proposed approach obtains good results in comparison to solutions obtained by GAMS. Additionally, in the Ref. [47], a hybrid algorithm for the capacitated multiperiod multicommodity lot-sizing problem was presented. In that problem, many commodities competed for the space and limited resources at each period. The proposed solution approach considers a Lagrangian relaxation and applies a metaheuristic based on XE for the uncapacitated version of the problem.

Additionally, XE has been applied to solve facility location problems modeled as bi-level programs. For example, in the Ref. [48] a bi-level problem was studied. Therein, the upper level is associated with a company that locates capacitated facilities aiming to minimize location and distribution costs. In the lower level, customers seek to maximize their preferences of being allocated to their most preferred facilities. A XE algorithm is designed to obtain upper-level solutions, while the lower level is solved through three different approaches via an adaptive random procedure, via a procedure based on a regret

cost, and by an exact method. The XE algorithm builds a set of upper-level solutions and for each of them, the resulting lower-level problem is solved. In other words, the classical nested approach is used, of algorithms designed to solve bi-level problems.

There exist other applications of XE in the context of multi-objective programming. In the Ref. [55] it was used to find efficient frontiers for problems with multiple multimodal objective functions of continuous variables. To obtain the Pareto fronts, the set of non-dominated solutions is divided in groups and a XE procedure is applied to each of them. That method is very similar to that proposed by the Ref. [56]. In the Ref. [57], an adapted XE method for estimating the Pareto front was proposed. To achieve the latter, a uniform grid in the space of the objectives is formed. Each cell in the grid contains its own elite population (with non-dominated solutions). As a consequence, the associated probability vectors are updated, respectively. At the end of the process, the approximation of the Pareto front consists of the union of the set of non-dominated solutions in each cell. Recently, in the Ref. [50] a multi-objective XE method was designed to solve a sustainable food supply chain. When the solutions' set is generated, a dominant matrix is considered to identify the best solution.

As mentioned before, the problem studied in this research is to be approached by a bi-objective nested cross-entropy algorithm (BONXEA). To design the BONXEA, the XE methods proposed in the Refs. [48,57] are considered. In the former, a discrete bi-level facility location problem with capacities is studied; meanwhile, in the latter, different multi-objective problems are analyzed. To initialize our proposed BONXEA, a grid in the objectives space must be defined. Since we are considering two objective functions, a 3×3 grid is created. The number of cells in the grid corresponds to the number of objectives being considered plus one. The rationale is to find the best solutions for each objective and for the central region of the Pareto frontier. To construct the grid, the minimum and maximum values for each objective function are needed. Let F_1 , F_2 and F_3 be the objective functions considered in the BONXEA, where F_1 corresponds to the objective function associated to the distance between the vaccination points, F_2 is associated to the balance of inhabitants in each vaccination point, and F_3 is the weighted sum of the standardization of F_1 and F_2 .

After that, a population of N upper-level solutions (y) is created. For each of these solutions, the lower-level problem is optimally solved (nested attribute of the BONXEA). Now, the upper-level objective functions can be evaluated. Solutions in the initial population are grouped into three subpopulations, each of them associated with F_1 , F_2 and F_3 , respectively. Three samples of size d that contain the best solutions, in terms of the quality of the corresponding objective function, are formed. For each sample, the frequencies of locating a vaccination point are updated in the construction phase. Therefore, new subpopulations of solutions are created based on the updated frequency vectors for each of the three samples. The procedure is repeated until a maximum number of iterations is reached. Non-dominated solutions and their respective objective function values are stored, and given as the output of our algorithm.

A pseudocode of the BONXEA is presented in Algorithm 1.

Algorithm 1 BONXEA**Input:** $N, d, MaxIter$ **Output:** P

```

1:  $Iter \leftarrow 0$ 
2:  $P \leftarrow \emptyset$  (Pareto front)
3:  $\gamma^{(k)} \leftarrow 0.5$  (Initial frequencies)
4: Define a  $3 \times 3$  grid that bounds  $F_1$  and  $F_2$ 
5: while  $Iter < MaxIter$  do
6:   for  $i : 1..N$  do
7:      $\bar{y}_i \leftarrow$  Create an upper level solution based on  $\gamma^{(k)}$ 
8:      $y_i \leftarrow$  Repair an upper level solution( $\bar{y}_i$ )
9:      $x_i \leftarrow$  Optimally solve the lower level( $y_i$ )
10:    Evaluate  $F_1(x_i), F_2(y_i)$  and  $F_3(x_i, y_i)$ 
11:   end for
12:   for each non-empty cell  $c$  of the  $3 \times 3$  grid do
13:     Identify the  $d$  elite solutions
14:     Update frequency vector  $\gamma^{(k)}$ 
15:     Obtain  $P_c$  accordingly with  $F_1$  and  $F_2$ 
16:      $P \leftarrow P \cup P_c$ 
17:   end for
18:   Update  $P$  with the non-dominated points
19:    $Iter \leftarrow Iter + 1$ 
20: end while

```

4.1. BONXEA Description**Solution encoding**

The proposed algorithm explores upper-level solutions, and for each of them, the optimal reaction of the lower level is obtained. In this case, an upper-level solution at iteration k is represented by the binary vector $y^{(k)} = (y_1^{(k)}, \dots, y_{|I|}^{(k)})$, in which a 1 indicates that a VP is located at site $i \in I$, and 0 otherwise.

Creation of solutions

At iteration k , N random solutions are created but based on a frequency vector $\gamma^{(k)} = (p_1^{(k)}, \dots, p_{|I|}^{(k)})$, that indicates the probability of locating a facility at each site. At the beginning of the BONXEA, when $k = 1$, probabilities are set as 0.5, that is, $\gamma^{(1)} = (0.5, \dots, 0.5)$. As the algorithm continues, frequency vectors $\gamma^{(k)}$ are going to be updated from one iteration to the next. That will lead us to favoring the facilities more convenient to locating VPs.

Repair of solutions

In the case when an upper-level solution $y^{(k)} = (y_1^{(k)}, \dots, y_{|I|}^{(k)})$ at iteration k has not located the p required VPs, a specific routine is performed to repair the solution, that is, to achieve feasibility. Specifically, the non-located VP associated with the higher frequency in vector $k = (p_1^{(k)}, \dots, p_{|I|}^{(k)})$ is located. If more than one VP needs to be located to achieve the p required, the repair procedure continues in an analogous manner. The repairing procedure stops when p VPs are located.

Optimal solution of the lower-level problem

Recall that once the located vaccination points are known, inhabitants are optimally allocated to these facilities. Allocation must take into account inhabitants' preferences in regard to the VPs. Since capacity at the VPs is not being taken into consideration, all the inhabitants may be allocated to their most preferred located facilities.

Therefore, the lower-level problem may be optimally solved by using an auxiliary matrix of ordered preferences, in which the indices of the facilities are ordered in a decreasing manner based on the preferences of each inhabitant. That is, each column of the auxiliary matrix contains the ordered indices of the VPs. Once the auxiliary matrix is

constructed, the lower-level problem can be optimally solved by using an easy procedure. For each inhabitant, their most preferred located VP is identified, and the allocation is made. By following the procedure described, the optimal allocation of inhabitants to VPs is achieved (see the Refs. [13,16]). Hence, an optimal solution of the lower level is obtained for a given upper-level fixed solution, that is, $x^{*(k)}(y^{(k)})$. For sake of simplicity, it is denoted by $x^{(k)}$.

Evaluation of the objective functions

Once an upper-level solution $y_i^{(k)}$ and its corresponding lower-level optimal response $x^{(k)}$ are obtained, the objective functions considered in the BONXEA can be evaluated, that is, F_1, F_2 and F_3 . Recall that F_1 is given by Equation (1), F_2 is given by Equation (3), and F_3 is given by $\alpha NF_1 + (1 - \alpha)NF_2$, where NF_1 and NF_2 are the standardization of F_1 and F_2 , respectively.

Creation of the subpopulations based on dominance

Firstly, we take the elite populations for each of the three cells considered in the 3×3 grid. Each elite population is associated to each one of the three objective functions considered, that is, F_1, F_2 and F_3 . Therefore, the best d solutions regarding objective F_1 are selected, and the objective F_2 is evaluated for such solutions. The non-dominated solutions in that cell are stored, which are denoted by $y^{1(k)}, y^{2(k)}, \dots, y^{d(k)}$. After that, the same procedure is repeated to identify the non-dominated solutions in the remaining two cells, that is, regarding F_2 and F_3 , respectively. The authors emphasize that to identify the non-dominated solutions, only objectives F_1 and F_2 are used because they are the real objectives in the problem under study. Recall that objective F_3 is an auxiliary objective function to identify the central part of the Pareto front.

Under this approach, three Pareto fronts are obtained, one for each of the three cells considered in the grid. Likewise, each subpopulation evolves independently from the other two subpopulations.

Updating the probabilities vector γ

At iteration k , vectors $\gamma_l^{(k)}$, with $l = 1, 2, 3$, are vectors that contain the frequency in which a specific VP is located in the solutions of the corresponding elite subpopulation performed, n . The reason for the use of vectors $\gamma_l^{(k)}$ is to store information of the components of the elite solutions, that is, to identify the repetition of VPs in the good-quality solutions. This is performed so as to favor locating these VPs in the next random-biased solutions. For each subpopulation l , the probabilities are obtained as follows:

$$\gamma_l^{(k)} = \left(\frac{\sum_{s=1}^d y_1^{s(k)}}{d}, \frac{\sum_{s=1}^d y_2^{s(k)}}{d}, \dots, \frac{\sum_{s=1}^d y_{|I|}^{s(k)}}{d} \right)$$

Creation of new solutions

To generate new solutions of each subpopulation l at iteration k , vectors $\gamma_l^{(k)}$ are used to determine whether a VP is located or not. As mentioned before, the procedure seeks to favor the more convenient VPs regarding the information provided by the elite solutions of each subpopulation. For the new created solutions, a random vector of size $|I|$ is generated, and if the random number associated to the i -th position is less than its corresponding probability $\gamma_{il}^{(k)}$, then that VP is located.

If a 0 appears in one position within vectors γ_l , then it means that in the new created solutions, a VP must not be located in that specific site. Otherwise, when a positive probability appears, the possibility exists to locate a VP at that potential site. In the case where the probability is 1, a VP must be located in that specific site for all the new created solutions. By following this criterion, the algorithm iteratively converges to good-quality solutions.

Stopping criterion

The stopping criteria are established as a fixed maximum number of iterations, and a predetermined number of iterations without updating the non-dominated set of solutions at each subpopulation.

Final Pareto front

At each iteration of the BONXEA, a Pareto front for each one of the three subpopulations is obtained. Furthermore, the obtained frontiers for each cell of the grid are updated at each iteration taking the previous fronts.

At the end of the BONXEA, the non-dominated sets of points obtained for each elite subpopulation are merged to form the final Pareto front. Some non-dominated solutions in a cell are dominated by some solutions of the Pareto front of another cell. Hence, at the end, only the non-dominated solutions of all the three cells are reported.

4.2. Computational Complexity of BONXEA

BONXEA starts by generating an initial population of N solutions of size $|I|$. Additionally, a repairing phase may be needed if less than p VPs are located. In the worst case, p insertions from all the $|I|$ possibilities are performed. Then, the lower level is optimally solved via an allocation algorithm in $O(|I| \times |J|)$ time.

After that, a sample of d elite solutions is selected for each of the c cells in the 3×3 grid, which incurs in $O(d \times c \times |I|)$ time. The evaluation of the first objective function requires $O(d \times c \times |I| \times |J|)$ time. The update of the probability vector requires the same time as the selection of the sample, that is, $O(d \times c \times |I|)$.

The algorithm repeats its main cycle for a predefined number $MaxIter$ of iterations. Hence, the computational complexity is $MaxIter$ times the sum of all the computational complexities identified above. In summary, the computational complexity of BONXEA is $O(d \times c \times |I| \times |J| \times MaxIter)$, which is of polynomial time.

5. Experimental Results

To measure the performance of the proposed algorithm and its ability to approximate the Pareto front, an intensive computational experimentation was carried out on a set of instances adapted from the literature. All computational experiments were performed on a computer with a 3.60 GHz Intel Core i7-4790 processor with 32.00 GB of RAM running Windows 8.1. The BONXEA code was coded with FICO Xpress 8.11 software.

The set of instances was adapted from the instances used for a discrete facility location problem with balanced customer allocation, which was proposed in the Ref. [7]. In that problem, the parameters regarding the instance size were included, that is, number of possible facilities and number of clients (VPs and inhabitants in our case, respectively). In addition, the distances between VPs and inhabitants were included. The missing parameters that are needed for our problem have been filled in as is described next. In particular, data related to the preferences of the inhabitants towards VPs were missing. These preferences were generated through the process described in the Ref. [58]. Basically, this process takes into account the existing costs, and generates modified fictitious costs to rank them in ascending order. Based on this, preferences are generated, ensuring that the closest VP is not always the most preferred by an inhabitant. It is worthwhile mentioning that under these instances' construction processes, all the generated instances are feasible for our problem.

For the problem studied in this research, the instances must contain the data on the number of VPs, the number of inhabitants, the predefined number of VPs that must be located, the preferences and the distances of the inhabitants towards the VPs, respectively.

We denote an instance as follows: $m - n - p - v$, as in the Ref. [7]. For example, 30-50-3-10 means that there are 30 potential VPs, 50 inhabitants, 3 VPs to be opened, and only 10 distance pairs that have been exchanged. The latter refers to a parameter that adds complexity to the instance.

In total, 45 different instances were generated by combining different numbers of the following parameters: (i) VPs to be opened ($|I|$), which were set from the set $\{20, 30, 50, 100\}$;

(ii) the number of inhabitants, which vary between $\{20, 30, 50, 100\}$; (iii) the number of VPs to open p , which were among $\{3, 4, 6, 10\}$; and (iv) the perturbation level v , which was used to modify the values of an instance without losing the structure, those values were chosen from $\{10, 30, 50, 100\}$.

Due to the stochasticity inherent in the BONXEA, 100 runs were performed for each instance. A summary of the results obtained from the experimentation is shown in Table 2. The first column indicates the label of the instance to which we refer. The remaining columns are associated with the non-dominated solutions. The second and fourth columns show the minimum and maximum number of non-dominated solutions obtained by BONXEA, respectively. The third column shows the average of the obtained non-dominated solutions among the 100 runs of the algorithm. The fifth and sixth columns show the minimum and maximum values for the objective function F_1 , respectively. Recall that the said objective function refers to the balance of the inhabitants in the open VPs. Finally, the minimum and maximum values obtained for the second objective function F_2 are shown in the seventh and eighth columns. It should be noted that the second objective function seeks to maximize the minimum distance between two open VPs. With these last four columns, we can identify the extremes of the obtained approximated Pareto front.

From Table 2 it can be seen that the BONXEA always obtained non-dominated solutions. The largest number of non-dominated solutions that was obtained is eight solutions (see instance 50-100-6-50). On average, there are at least two non-dominated solutions for each instance. We note that F_1 refers to the balancing of inhabitants. In a case of perfect balancing of inhabitants among the VPs, $F_1 = 0$. If this is not possible, the range in which F_1 varies is very limited. That is, there are not many non-dominated solutions with large F_1 values. For example, for the instances 30-30-3-1 and 30-30-3-100, a perfect balance is found, that is, there are exactly 10 inhabitants that go to each open VP. However, instance 100-100-3-10 shows a surprising result. A total of 100 clients were considered and the worst reported balance was 49. This implies that there is a lot of disparity between the crowding in the open VPs. However, after reviewing this solution, we realized that the balance of 49 corresponds to the best value for F_2 . That is, very distant VPs cause the inhabitants to be mostly concentrated in a central VP. That occurs only in that particular instance, and due to the spatial distribution of the inhabitants and potential VPs. It is also important to clarify that the obtained values of F_2 are integers because that is the manner in which they are defined in the data, and Euclidean distances between inhabitants and VPs are not being calculated.

From the results obtained, we also identify that the proposed algorithm finds solutions that allow us to correctly approximate the Pareto fronts of this hierarchized problem. To illustrate the above, some of the obtained Pareto fronts were randomly selected and have been plotted in Figure 1.

It can be seen in Figure 1 that there is an adequate number of solutions plotted. In the graphics presented, the maximum number of solutions obtained in the Pareto front is 9 and the minimum is 6. However, as already mentioned above, it is very important to take into account that the first leader's objective function (F_1) seeks to minimize balance among inhabitants that attend to a specific open VP, which is a small integer value, so excessive possible values for that target do not exist.

In addition, these graphs show the 3×3 grid that is carried out in each iteration to consider the best solutions for the creation of the nine sub-populations of solutions. Within the grid, it is observed that in the specific grid that corresponds to the best values for F_1 and F_2 , the number of solutions contained is from one to three. This validates the idea of considering the grid to always find the best solution for each objective function. At least in the 45 instances considered in this computational experimentation, this behavior is maintained. Likewise, it can be observed that there is diversity within the solutions, that is, some are better for F_1 , others better for F_2 , and some others that are in the center, according to the weighted sum considering the objective function.

Table 2. Summary of the solutions obtained by the BONXEA.

Instance	Min	Avg	Max	Min F_1	Max F_1	Min F_2	Max F_2
20-20-3-1	2	3.5	4	1	4	29	37
20-20-3-10	1	2.6	4	1	9	16	18
20-20-3-50	2	2	2	1	17	7	18
20-20-6-1	2	2.5	4	1	3	9	13
20-20-6-10	3	3.5	5	1	5	7	14
20-20-6-50	2	2.9	4	1	4	5	10
30-30-3-1	2	4.1	7	0	9	48	71
30-30-3-10	2	2.9	4	8	17	26	28
30-30-3-100	2	3.6	5	0	8	23	26
30-30-4-10	2	3	4	10	13	10	27
30-30-4-100	2	3.2	6	1	4	20	26
30-50-3-10	1	2.1	4	1	17	4	28
30-50-3-50	2	2.9	4	1	6	25	27
30-50-3-100	1	2.9	4	4	14	1	26
30-50-6-10	3	3.5	5	1	8	11	23
30-50-6-50	3	4.1	6	2	14	15	22
30-50-6-100	2	4.7	6	1	13	6	22
30-50-10-10	3	3.7	5	5	6	14	19
30-50-10-50	2	3.4	4	2	7	7	14
30-50-10-100	2	3.5	5	4	8	7	13
30-100-3-50	2	4.1	6	4	10	16	28
30-100-3-100	2	3.4	6	4	12	25	28
30-100-6-10	3	4	6	4	12	10	24
30-100-6-50	2	4.6	7	3	21	9	23
30-100-6-100	4	5.2	7	3	13	8	22
30-100-10-10	2	3.5	5	9	16	9	19
30-100-10-50	2	3.7	5	7	15	9	18
30-100-10-100	3	4.7	7	6	15	4	18
50-50-3-10	1	2.5	6	2	18	1	41
50-50-3-50	3	3.7	5	1	5	36	48
50-50-3-100	1	2.8	5	1	1	47	47
50-50-6-10	3	4.6	6	1	11	26	39
50-50-10-10	2	3.3	5	2	7	21	39
50-50-10-50	2	3.7	6	2	8	26	33
50-50-10-100	2	3.3	6	4	12	2	34
50-100-3-10	3	3.5	6	1	14	44	48
50-100-3-50	2	3.9	7	1	12	37	47
50-100-3-100	2	4.6	7	7	23	46	47
50-100-6-50	3	4.2	8	3	16	31	39
50-100-6-100	2	4	5	3	12	33	42
50-100-10-50	2	3.3	4	2	6	23	32
50-100-10-100	3	4	5	3	12	27	35
100-100-3-1	1	2.6	4	1	5	102	105
100-100-3-10	1	3.5	8	1	49	3	95
100-100-3-50	2	3.8	5	1	20	93	97

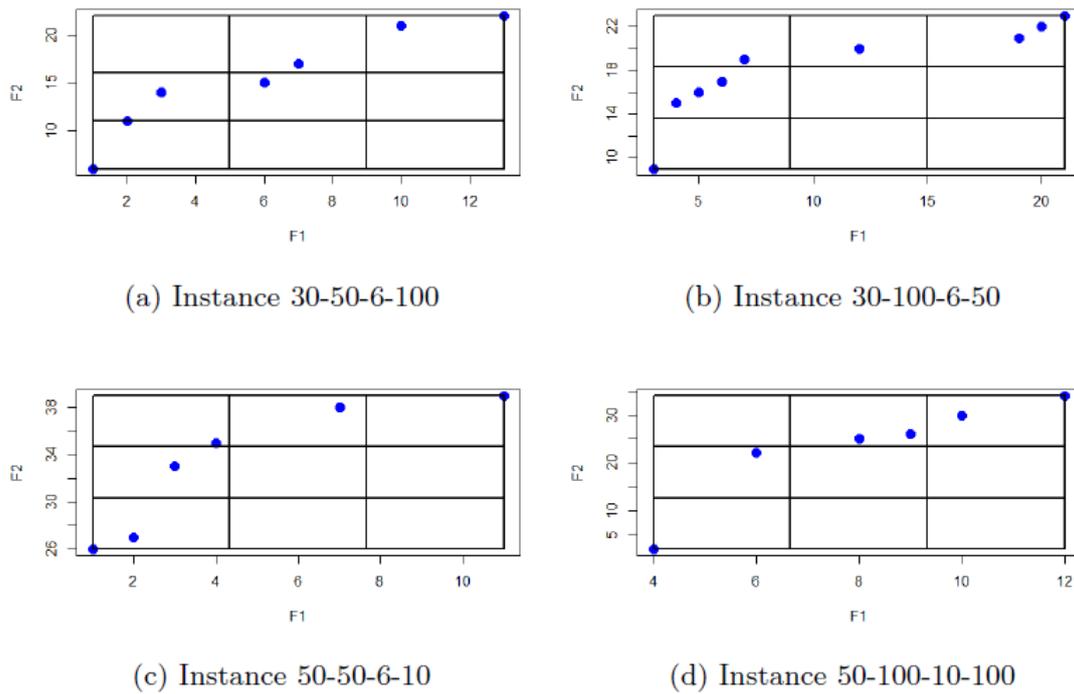


Figure 1. Selected approximated Pareto fronts obtained by the BONXEA.

In order to illustrate the limited space in which the leader’s objective function is embedded (regarding the balance), one independent run of the BONXEA was performed and 100 runs were set as the stopping criterion. Therefore, we registered the best value obtained for F_1 at each iteration. This may give us an idea of the convergence capability of the BONXEA. It can be seen in Figure 2 that from a balance value of 4 it improves to 2, and no further improvement is reported in the remaining 90 iterations. Additionally, at each iteration of the BONXEA, other non-dominated solutions may arise, but this particular analysis is performed to demonstrate the convergence of the BONXEA regarding a particular leader’s objective function. In Figure 3, the same analysis is performed, but with respect to F_2 . It can also be noticed that the BONXEA obtains the best value of F_2 in the first 40 iterations. This is a good indicator that the BONXEA contains a well-designed intensification component.

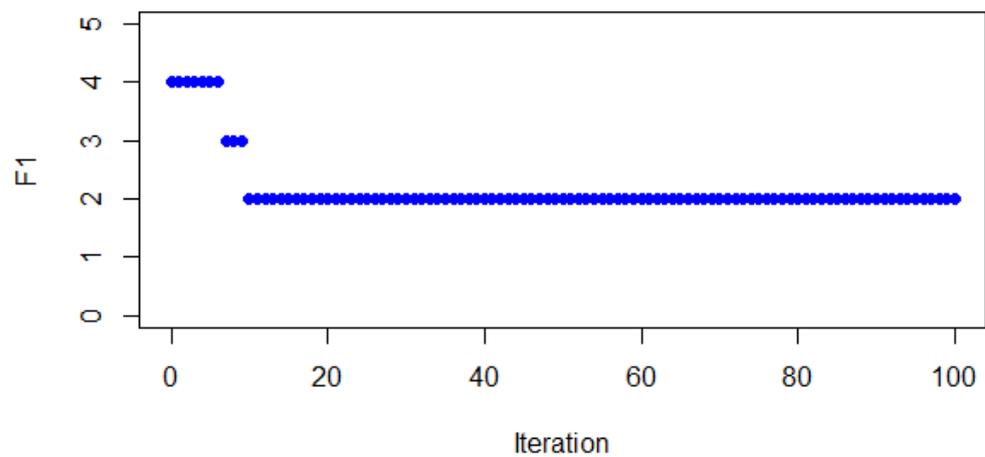


Figure 2. Illustrating the convergence of BONXEA regarding F_1 .

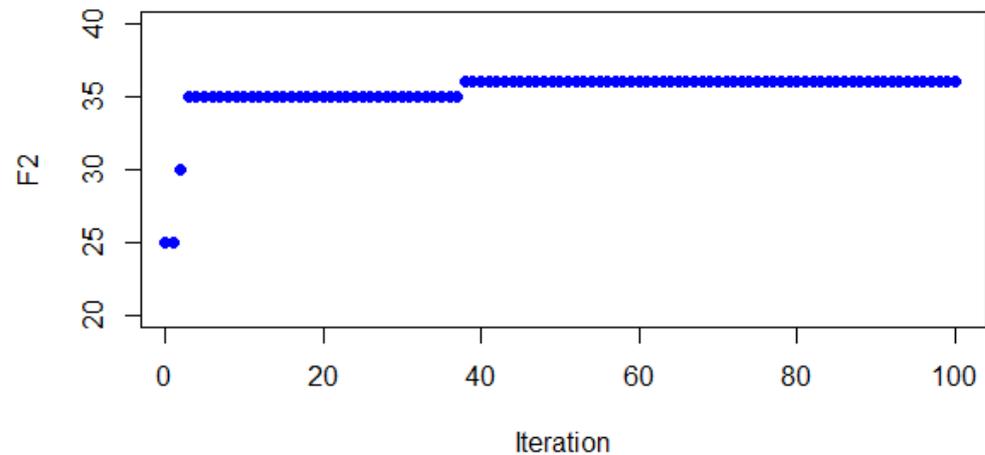


Figure 3. Illustrating the convergence of BONXEA regarding F_2 .

We may summarize that the proposed algorithm generates a reasonably-sized set of efficient solutions for our problem. Therefore, the decision-maker will have to choose the solution that is more convenient to him/her at the moment when the real problem is being solved. In other words, among all the non-dominated solutions obtained by the BONXEA, only one can be implemented in practice. However, this is common for bi-objective practical decision-making problems.

6. Conclusions and Further Research Directions

As mentioned throughout this manuscript, implementing measures to fight the COVID-19 pandemic is crucial. For example, social distancing (healthy distance) and avoiding crowds are globally accepted as good steps to prevent infections and to try to control the pandemic. It is in this context that a vaccination-point location problem that considers inhabitants' preferences arises. Its purpose is to contribute to maintaining the healthy distance between inhabitants who come to receive their vaccine against the COVID-19 virus. In this paper, the government sought to open vaccinations points as far from each other as possible, but at the same time, sought to have a balanced demand for vaccines at each open vaccination point. This is not straightforward because the government does not impose restrictions on the inhabitants regarding which vaccination point they can go to. On the contrary, inhabitants decide where to go based on their own preferences, which may be decided by distance, personal interests, and type of vaccine, among others. If the government assigns the inhabitants to the VPs in a mandatory manner, the closest assignment rule can be selected and the problem reduces its complexity.

The situation is modeled under a bi-objective bi-level programming approach. the government is associated with the leader, and the inhabitants, with the follower. As mentioned, the inhabitants' preferences are taken into account when they are allocated to the open vaccination points. Different constraints are considered, which complicate the problem.

To solve this problem, we proposed a metaheuristic approach that is capable of approximating the Pareto front in an effective manner. A bi-objective nested cross-entropy algorithm is designed. Both considered objective functions are taken into account in the algorithm. The pertinent mechanisms are proposed to identify the approximation of the Pareto front. Additionally, for each leader's solution, the parameterized follower's problem is optimally solved. This is the nested approach of the proposed algorithm. Therefore, initial solutions are randomly generated followed by a procedure that creates a grid to classify solutions. Then, the more convenient components of a solution are identified, with respect to the cell of the grid that the solution belongs to. Based on the latter, new solutions are generated following a components-biased manner. The algorithms continue until a stopping criterion is reached.

It can be seen from the computational experimentation that the proposed algorithm obtained a Pareto front in good shape. Insights can be gained from the obtained results. However, it is important to point out that the structure of the first leader's objective function does not have many values. Thus, few values may be obtained, which significantly reduces the number of solutions that may conform to the Pareto front. In the literature, to the best knowledge of the authors, there are no papers with all the characteristics considered in this study. As a consequence, this is the first bi-objective bi-level programming model for a problem with these assumptions. For this reason, there are no benchmark algorithms or instances that can be considered to compare our proposed algorithm.

A future research direction may be the application of this problem to a real case study. However, it is well-known that government agencies are not often open to sharing data or to allowing academic collaborations. A comparison against real implemented decisions would give us interesting managerial insights. As a second research direction, we may apply this model to a different context. Maybe in the private sector, a situation could arise in which a company has similar interests as the ones stated by the leader in our problem. Customers could be considered the inhabitants of our problem. Usually, it is important to allow customers to be free to choose the service of their preference.

Author Contributions: Conceptualization, E.S. and J.-F.C.-V.; methodology, E.S. and J.-F.C.-V.; software, E.S. and J.-F.C.-V.; validation, E.S., S.N.-G. and J.-F.C.-V.; formal analysis, E.S., S.N.-G. and J.-F.C.-V.; investigation, E.S., S.N.-G. and J.-F.C.-V.; resources, E.S. and J.-F.C.-V.; data curation, E.S. and J.-F.C.-V.; writing—original draft preparation, E.S., S.N.-G. and J.-F.C.-V.; writing—review and editing, S.N.-G. and J.-F.C.-V.; visualization, E.S.; supervision, J.-F.C.-V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data used in this research is available upon a respectful request, via email.

Acknowledgments: We sincerely thank Mohini Ganguly of the obex project for language editing support.

Conflicts of Interest: The authors have no conflict of interest to declare.

References

1. Duijzer, L.E.; van Jaarsveld, W.L.; Wallinga, J.; Dekker, R. Dose-optimal vaccine allocation over multiple populations. *Prod. Oper. Manag.* **2018**, *27*, 143–159. [[CrossRef](#)]
2. Daskin, M. Network and discrete location: Models, algorithms and applications. *J. Oper. Res. Soc.* **1997**, *48*, 763–764. [[CrossRef](#)]
3. Eiselt, H.A.; Marianov, V. *Foundations of Location Analysis*; Springer: Berlin/Heidelberg, Germany, 2011.
4. Laporte, G.; Nickel, S.; Saldanha-da Gama, F. *Introduction to Location Science*; Springer: Berlin/Heidelberg, Germany, 2019.
5. Dasarathy, B.; White, L.J. A maxmin location problem. *Oper. Res.* **1980**, *28*, 1385–1401. [[CrossRef](#)]
6. Erkut, E. The discrete p-dispersion problem. *Eur. J. Oper. Res.* **1990**, *46*, 48–60. [[CrossRef](#)]
7. Marín, A. The discrete facility location problem with balanced allocation of customers. *Eur. J. Oper. Res.* **2011**, *210*, 27–38. [[CrossRef](#)]
8. McVitie, D.G.; Wilson, L.B. The stable marriage problem. *Commun. ACM* **1971**, *14*, 486–490. [[CrossRef](#)]
9. Iwama, K.; Miyazaki, S. A survey of the stable marriage problem and its variants. In Proceedings of the International Conference on Informatics Education and Research for Knowledge-Circulating Society (ICKS 2008), Kyoto, Japan, 17 January 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 131–136.
10. Hanjoul, P.; Peeters, D. A facility location problem with clients' preference orderings. *Reg. Sci. Urban Econ.* **1987**, *17*, 451–473. [[CrossRef](#)]
11. Cánovas, L.; García, S.; Labbé, M.; Marín, A. A strengthened formulation for the simple plant location problem with order. *Oper. Res. Lett.* **2007**, *35*, 141–150. [[CrossRef](#)]
12. Hansen, P.; Kochetov, Y.; Mladenovi, N. *Lower Bounds for the Uncapacitated Facility Location Problem with User Preferences*; Groupe d'études et de recherche en analyse des décisions; HEC Montréal: Montréal, QC, Canada, 2004.
13. Marić, M.; Stanimirović, Z.; Milenković, N. Metaheuristic methods for solving the bilevel uncapacitated facility location problem with clients' preferences. *Electron. Notes Discret. Math.* **2012**, *39*, 43–50. [[CrossRef](#)]
14. Maldonado-Pinto, S.; Casas-Ramírez, M.S.; Camacho-Vallejo, J.F. Analyzing the performance of a hybrid heuristic for solving a bilevel location problem under different approaches to tackle the lower level. *Math. Probl. Eng.* **2016**, *2016*, 9109824. [[CrossRef](#)]

15. Casas-Ramírez, M.S.; Camacho-Vallejo, J.F.; Díaz, J.A.; Luna, D.E. A bi-level maximal covering location problem. *Oper. Res.* **2020**, *20*, 827–855. [[CrossRef](#)]
16. Calvete, H.I.; Galé, C.; Iranzo, J.A.; Camacho-Vallejo, J.F.; Casas-Ramírez, M.S. A matheuristic for solving the bilevel approach of the facility location problem with cardinality constraints and preferences. *Comput. Oper. Res.* **2020**, *124*, 105066. [[CrossRef](#)]
17. Camacho-Vallejo, J.F.; García-Vélez, J.C.; Corpus, C. A 5G Hubs Location Hierarchized Problem that Balances the Connection of the Users. *Mob. Netw. Appl.* **2022**, 1–11. [[CrossRef](#)]
18. Rahman, T.; Taghikhah, F.; Paul, S.K.; Shukla, N.; Agarwal, R. An agent-based model for supply chain recovery in the wake of the COVID-19 pandemic. *Comput. Ind. Eng.* **2021**, *158*, 107401. [[CrossRef](#)]
19. Burgos, D.; Ivanov, D. Food retail supply chain resilience and the COVID-19 pandemic: A digital twin-based impact analysis and improvement directions. *Transp. Res. Part E Logist. Transp. Rev.* **2021**, *152*, 102412. [[CrossRef](#)]
20. Chowdhury, P.; Paul, S.K.; Kaisar, S.; Moktadir, M.A. COVID-19 pandemic related supply chain studies: A systematic review. *Transp. Res. Part E Logist. Transp. Rev.* **2021**, *148*, 102271. [[CrossRef](#)]
21. Sun, L.; DePuy, G.W.; Evans, G.W. Multi-objective optimization models for patient allocation during a pandemic influenza outbreak. *Comput. Oper. Res.* **2014**, *51*, 350–359. [[CrossRef](#)]
22. Luo, L.; Wan, X.; Wang, Q. A multi-period location-allocation model for integrated management of emergency medical supplies and infected patients during epidemics. *Comput. Ind. Eng.* **2022**, *173*, 108640. [[CrossRef](#)]
23. Sarkar, S.; Pramanik, A.; Maiti, J.; Reniers, G. COVID-19 outbreak: A data-driven optimization model for allocation of patients. *Comput. Ind. Eng.* **2021**, *161*, 107675. [[CrossRef](#)]
24. Buhat, C.A.H.; Duero, J.C.C.; Felix, E.F.O.; Rabajante, J.F.; Mamplata, J.B. Optimal allocation of COVID-19 test kits among accredited testing centers in the Philippines. *J. Healthc. Inform. Res.* **2021**, *5*, 54–69. [[CrossRef](#)]
25. Thul, L.; Powell, W. Stochastic optimization for vaccine and testing kit allocation for the COVID-19 pandemic. *Eur. J. Oper. Res.* **2023**, *304*, 325–338. [[CrossRef](#)]
26. Parvin, H.; Beygi, S.; Helm, J.E.; Larson, P.S.; Van Oyen, M.P. Distribution of medication considering information, transshipment, and clustering: Malaria in Malawi. *Prod. Oper. Manag.* **2018**, *27*, 774–797. [[CrossRef](#)]
27. Abdin, A.F.; Fang, Y.P.; Caunhye, A.; Alem, D.; Barros, A.; Zio, E. An optimization model for planning testing and control strategies to limit the spread of a pandemic—The case of COVID-19. *Eur. J. Oper. Res.* **2023**, *304*, 308–324. [[CrossRef](#)]
28. Ozdemir, I.; Dursunoglu, C.F.; Kara, B.Y.; Dora, M. Logistics of temporary testing centers for coronavirus disease. *Transp. Res. Part C Emerg. Technol.* **2022**, *145*, 103954. [[CrossRef](#)]
29. Manupati, V.K.; Schoenherr, T.; Wagner, S.M.; Soni, B.; Panigrahi, S.; Ramkumar, M. Convalescent plasma bank facility location-allocation problem for COVID-19. *Transp. Res. Part E Logist. Transp. Rev.* **2021**, *156*, 102517. [[CrossRef](#)]
30. Mitropoulos, P.; Mitropoulos, I.; Giannikos, I.; Sissouras, A. A biobjective model for the locational planning of hospitals and health centers. *Health Care Manag. Sci.* **2006**, *9*, 171–179. [[CrossRef](#)]
31. Duijzer, L.E.; Van Jaarsveld, W.; Dekker, R. Literature review: The vaccine supply chain. *Eur. J. Oper. Res.* **2018**, *268*, 174–192. [[CrossRef](#)]
32. Enayati, S.; Özaltın, O.Y. Optimal influenza vaccine distribution with equity. *Eur. J. Oper. Res.* **2020**, *283*, 714–725. [[CrossRef](#)]
33. Medlock, J.; Galvani, A.P. Optimizing influenza vaccine distribution. *Science* **2009**, *325*, 1705–1708. [[CrossRef](#)]
34. Mohammadi, M.; Dehghan, M.; Pirayesh, A.; Dolgui, A. Bi-objective optimization of a stochastic resilient vaccine distribution network in the context of the COVID-19 pandemic. *Omega* **2022**, *113*, 102725. [[CrossRef](#)]
35. Rastegar, M.; Tavana, M.; Meraj, A.; Mina, H. An inventory-location optimization model for equitable influenza vaccine distribution in developing countries during the COVID-19 pandemic. *Vaccine* **2021**, *39*, 495–504. [[CrossRef](#)]
36. Bertsimas, D.; Digalakis, V., Jr.; Jacquillat, A.; Li, M.L.; Previero, A. Where to locate COVID-19 mass vaccination facilities? *Nav. Res. Logist. (NRL)* **2022**, *69*, 179–200. [[CrossRef](#)]
37. Wang, S.; Wu, Y.J.; Li, R. An Improved Genetic Algorithm for Location Allocation Problem with Grey Theory in Public Health Emergencies. *Int. J. Environ. Res. Public Health* **2022**, *19*, 9752. [[CrossRef](#)]
38. Yarmand, H.; Ivy, J.S.; Denton, B.; Lloyd, A.L. Optimal two-phase vaccine allocation to geographically different regions under uncertainty. *Eur. J. Oper. Res.* **2014**, *233*, 208–219. [[CrossRef](#)]
39. Abbasi, B.; Fadaki, M.; Kokshagina, O.; Saeed, N.; Chhetri, P. Modeling Vaccine Allocations in the COVID-19 Pandemic: A Case Study in Australia. Available online: <https://ssrn.com/abstract=3744520> (accessed on 26 June 2022). [[CrossRef](#)]
40. Tang, L.; Li, Y.; Bai, D.; Liu, T.; Coelho, L.C. Bi-objective optimization for a multi-period COVID-19 vaccination planning problem. *Omega* **2022**, *110*, 102617. [[CrossRef](#)]
41. Li, X.; Pan, Y.; Jiang, S.; Huang, Q.; Chen, Z.; Zhang, M.; Zhang, Z. Locate vaccination stations considering travel distance, operational cost, and work schedule. *Omega* **2021**, *101*, 102236. [[CrossRef](#)]
42. Dempe, S. *Foundations of Bilevel Programming*; Springer Science & Business Media: Berlin, Germany, 2002.
43. Colson, B.; Marcotte, P.; Savard, G. An overview of bilevel optimization. *Ann. Oper. Res.* **2007**, *153*, 235–256. [[CrossRef](#)]
44. Plastria, F. A note towards improved homeland defense. *Omega* **2012**, *40*, 244–248. [[CrossRef](#)]
45. Rubinstein, R.Y. Optimization of computer simulation models with rare events. *Eur. J. Oper. Res.* **1997**, *99*, 89–112. [[CrossRef](#)]
46. Caserta, M.; Rico, E.Q. A cross-entropy-Lagrangian hybrid algorithm for the multi-item capacitated lot-sizing problem with setup times. *Comput. Oper. Res.* **2009**, *36*, 530–548. [[CrossRef](#)]

47. Hosseini-zhad, S.J.; Salhi, S.; Jabalameli, M.S. A cross-entropy-based heuristic for the capacitated multi-source weber problem with facility fixed cost. *Comput. Ind. Eng.* **2015**, *83*, 151–158. [[CrossRef](#)]
48. Casas-Ramírez, M.S.; Camacho-Vallejo, J.F.; Martínez-Salazar, I.A. Approximating solutions to a bilevel capacitated facility location problem with customer's patronization toward a list of preferences. *Appl. Math. Comput.* **2018**, *319*, 369–386. [[CrossRef](#)]
49. Wang, Z.; Soleimani, H.; Kannan, D.; Xu, L. Advanced cross-entropy in closed-loop supply chain planning. *J. Clean. Prod.* **2016**, *135*, 201–213. [[CrossRef](#)]
50. Kalantari, F.; Hosseini-zhad, S.J. A Multi-objective cross-entropy-based algorithm for sustainable global food supply chain with risk considerations: A case study. *Comput. Ind. Eng.* **2022**, *164*, 107766. [[CrossRef](#)]
51. Santosa, B.; Budiman, M.A.; Wiratno, S.E. A cross-entropy-genetic algorithm for m-machines no-wait job-shopscheduling problem. *J. Intell. Learn. Syst. Appl.* **2011**, *3*, 171. [[CrossRef](#)]
52. Ning, X.; Li, P. A cross-entropy approach to the single row facility layout problem. *Int. J. Prod. Res.* **2018**, *56*, 3781–3794. [[CrossRef](#)]
53. Perelman, L.; Ostfeld, A. An adaptive heuristic cross-entropy algorithm for optimal design of water distribution systems. *Eng. Optim.* **2007**, *39*, 413–428. [[CrossRef](#)]
54. De Boer, P.T.; Kroese, D.P.; Mannor, S.; Rubinstein, R.Y. A tutorial on the cross-entropy method. *Ann. Oper. Res.* **2005**, *134*, 19–67. [[CrossRef](#)]
55. Unveren, A.; Acan, A. Multi-objective optimization with cross-entropy method: Stochastic learning with clustered pareto fronts. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation, Singapore, 25–28 September 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 3065–3071.
56. Kroese, D.P.; Porotsky, S.; Rubinstein, R.Y. The cross-entropy method for continuous multi-extremal optimization. *Methodol. Comput. Appl. Probab.* **2006**, *8*, 383–407. [[CrossRef](#)]
57. Caballero, R.; Hernández-Díaz, A.G.; Laguna, M.; Molina, J. Cross-entropy for multiobjective combinatorial optimization problems with linear relaxations. *Eur. J. Oper. Res.* **2015**, *243*, 362–368. [[CrossRef](#)]
58. Casas-Ramírez, M.S.; Camacho-Vallejo, J.F. Solving the p-median bilevel problem with order through a hybrid heuristic. *Appl. Soft Comput.* **2017**, *60*, 73–86. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.