A Novel Robust Model for Health Care Facilities Location-Allocation Considering Pre Disaster and Post Disaster Characteristics

Author(s): Mahdi Alinaghian\textsuperscript{1*}, seyed reza Hejazi\textsuperscript{2}, Nooshin Bajoul\textsuperscript{3}, kosar sadeghi velni\textsuperscript{4}

1.*Department of Industrial and Systems Engineering, Isfahan University of Technology, 84156-83111 Isfahan, Iran. 031-33915511, +989127920346, alinaghian@iut.ac.ir

2. Department of Industrial and Systems Engineering, Isfahan University of Technology, 84156-83111 Isfahan, Iran. +989131029092, rehejazi@iut.ac.ir

3. Department of Industrial and Systems Engineering, Isfahan University of Technology, 84156-83111 Isfahan, Iran. +989132198615, n.bajool@gmail.com

4. Department of Industrial Engineering, Fouman Faculty of Engineering, University of Tehran, 013-34915103 Fouman, Iran. +989113411121, Sadeghi.kosar@ut.ac.ir

Abstract
Health care centers are one of the most important municipal facilities that are involved in providing personal and social health. Locating medical centers is an important strategy to improve these facilities performance and reduce the relief process delay. On the other hand, during disasters, these centers have a critical role in the human casualty's reduction. In this paper, a new Robust Mathematical Modeling is simultaneously provided to locate and allocate healthcare facilities, including different service levels of medical care, concerning the characteristics in normal and disaster situations. Furthermore, to help victims and prevent overcrowding in medical centers during disasters, the establishment of temporary and outpatient centers is allowed for emergency provision of basic services. Also, the possibility to send equipment and medical teams to these centers is considered. To solve the problem, two metaheuristic algorithms, harmony search algorithm and hybrid Tabu search combined with variable neighborhood search algorithm, and a lower bound based on Lagrangian relaxation method are presented. To examine the proposed algorithms, sample problems are randomly generated. The results of exact solutions and the lower bound resulting from the Lagrangian relaxation method are compared with the meta-heuristic algorithms results. The results show the good performance of the algorithms.

Keywords: Health management system; Network reliability; Hierarchical Healthcare Facility Location Problem; Robust Optimization; Variable Neighborhood Search Algorithm; Harmony Search Algorithm.

1. Introduction
Healthcare centers are one of the most urban important facilities that are directly involved in providing personal and social health. Therefore, fast, timely, and cheap access to these centers in every community is important, especially in urban communities. On the other hand, every year,
natural disasters such as earthquakes, floods, storms... catch various parts of the world. Iran is no exception in this regard, and according to studies from 1980 to 2008, 138 similar disasters occurred in this country. As a result, the annual average of 2689 deaths has occurred and about $737 million damage incurred to the country [1]. The increasing number of disasters and the increase in their destructive range on the one hand, and population growth, on the other hand, lead to an increase in material damage and human loss in such events. Despite technological advances, the sufferings caused by these events have always been one of the main obstacles to the sustainable development of countries. Although the damage caused by these events is not compensated in various ways, using preventive measures and proper planning for being ready to deal with these events, we can reduce losses to the minimum level. One of the most important measures is proper planning during the establishment of healthcare facilities. Since the severity and magnitude of these events are often large, after these events, requests for the rescue operation are also quite extensive, and the relief centers that are able to satisfy the needs of the urban centers in normal circumstances are not often quick to respond to the requests in this case. In addition, it is noteworthy that dealing with victims, providing medical aids, and transferring the injured people to relief centers at the right time, especially in the first 72 hours after the disaster (Relief Golden Hours), play a very important role in reducing deaths and disabilities caused by disasters [2]. Healthcare facilities are generally hierarchical so that in any hierarchy, a series of services are provided, and based on the type and quantity of services provided, these centers have different levels of classification. In this paper, it is assumed that higher levels of health-care providers cover all services provided at the lower levels. The random and unpredictable nature of natural disasters (especially earthquakes) requires a comprehensive disaster management plan to reduce the risks and relieve the suffering caused by the disaster. In fact, the main objective of responses and relief reactions is planning to better deal with such phenomena and public awareness to reduce deaths, injuries and loss of property [3]. Because of the nature of disaster to plan for locating the treatment centers, taking into account the uncertainties in the data is inevitable. Lack of attention to the uncertainty of the disaster may lead to improper planning, and it can have a lot of facilities damaged during a disaster or be far away from disaster centers. This can make it difficult to provide services to victims. On the other hand, in the event of a disaster, the establishment of temporary relief centers is recommended. This is because these centers, on the one hand, reduce the time of initial relief services to the injured people and, on the other hand, only those who need higher levels of treatment are transferred to healthcare facilities which are on a higher level. This reduces the transfer rate, which is very important during disasters, due to the restrictions on transferring injured people. The higher levels reduce congestion in permanent healthcare facilities, which can provide better services in the centers. On the other hand, while the temporary relief centers according to the human resources capacity of the relief are also important, the possibility to send relief teams to temporary centers through hospitals and other permanent centers should be considered. In short, the innovations presented in this paper include below:
1. Provide a robust model to locate hierarchical healthcare facilities in disaster conditions,
2. Take into account the possibility of establishing temporary relief centers and transferring medical service teams from permanent medical centers to temporary ones,
3. Take into account the possibility of transferring patients from lower-level centers to medical care that are at higher levels. In other words, the model has referral property.
4. Provide two proposed meta-heuristic methods based on the harmony search algorithm, and Tabu search hybrid algorithm with the variable neighborhood search algorithm,
5. Provide a lower bound based on the Lagrangian relaxation method for the proposed problem.
6. Finally, to examine the proposed algorithms, a number of sample problems are randomly generated in small and large sizes and the results of exact solutions and the lower bound resulting from the Lagrangian relaxation method are evaluated and compared with the results of the meta-heuristic algorithms. The results show the good performance of the proposed algorithms.

2. Literature review

In this section, the studies were conducted in three areas of locating healthcare facilities regardless of hierarchy, locating hierarchical healthcare facilities, and locating facilities in disasters.

2.1. Location models of healthcare facilities without considering hierarchy

The healthcare location literature has tended to address three major topics, which we refer to as accessibility, adaptability, and reliability.

2.1.1 Accessibility

It means the ability of patients and customers to access healthcare centers or, in the case of emergency services, of relief workers and relief services to access patients. Generally, input parameters in this type of model such as cost, demand, travel distance or time are considered as fixed non-random ones [4]. Eaton et al. used a maximal covering model to help planners in Austin, TX select permanent bases for their emergency medical services. This model is solved using the substitution and greedy adding algorithms [5]. Jacobs et al. used the P-median model with a limited capacity of facilities to optimize the collection, testing, and distribution of blood products in North Virginia and Carolina [6]. McAleer and Naqvi also used the P-median model for the relocation of ambulances in Belfast, Ireland. The problem was the establishment of four facilities to serve 54 demand points. To solve this problem, they used a heuristic approach and divided the demand points into four parts and placed each facility in possible locations respectively. Therefore, they reached a number of acceptable solutions in each part. They then examined all possible combinations of acceptable sites using the 54 demand points [7].

Osinuga et al. developed an integrated methodology in the field of healthcare location that minimizes the weighted average Euclidean distance between existing facilities and the new facility in a way that demands will not have to travel a very long distance to access medical facilities. Geographic information system (GIS) was used to store and handle the coordinates and weights of the existing facilities. However, the Weber model and Weiszfeld’s algorithm was employed to determine the new healthcare location. This technique applied to a case study in one of the local government areas in Nigeria [8].

Cardoso et al. proposed a multi-objective and multi-period mathematical programming model to support planning decisions in the Long-Term Care (LTC) sector. This model provides information to help plan the delivery of institutional LTC services in the medium-term both in terms of location selection and capacity planning when it is intended to move towards an equitable provision of care. For this purpose, they considered three equity-related objectives—equity of access, geographical equity and socioeconomic equity. The cost considerations were also taken into account. The model objective function incorporated multiple objectives. They illustrated the applicability of the model through the resolution of a case study in the Great Lisbon region in Portugal [9].

Dogan et al. focused on the problem of locating preventive healthcare facilities (PHC). The most important factors that increase participation in PHC programs include establishing an appropriate infrastructure and providing satisfactory healthcare. To this end, they used a multi-objective mixed-linear programming model to locate PHC facilities to ensure maximum participation and
provide timely services to potential customers. They applied a model for a case study to determine the location of cancer early detection, screening and education centers in Istanbul, Turkey [10].

2.1.2. Adaptability
The emphasis on this feature means that location decisions must be based on the uncertain conditions in the future, especially in the case of facilities such as hospitals, which can be difficult, if not impossible, to relocate as conditions change [4]. One of the common actions in this area is scenario-based planning. In scenario-based planning, some of the decisions are taken just before determining the right scenario, while other decisions are taken when the information was obtained from the right scenario. In the case of location, locating facilities generally occurs before knowing which scenario is chosen. The allocation of demand points to these places can normally occur after a scenario and knowing which scenario occurs. Three performance measures are used in scenario-based planning:
- Optimizing the expected performance
- Minimizing the worst-case performance
- Minimizing the worst-case regret

To deal with these issues, Daskin et al. designed a model to minimize the maximum regret among a subset of scenarios that have already been identified with a probability of at least $\alpha$ [11]. Carson and Batta considered the problem of locating an ambulance for State University at Buffalo, so that the model is scheduled for four different time periods according to change of daily conditions and the population at various time intervals [12]. Reveille et al. offered a number of coverage models so that the demand nodes, and also any facility established should be covered by different facilities and the model. The logic of the models is based on the fact that if a disaster occurs, emergency services in the area can be out of order or inaccessible, and therefore, they should be covered by facilities in other areas [13].

2.1.3. Reliability
Reliability is different from adaptability, because adaptability means the ability to function properly in uncertain conditions in the future. Uncertainty generally occurs for inputs such as demand and costs. However, reliability is the ability to function properly when a part of a system fails [14]. In other words, the ability to adapt is related to the system environment and reliability is the uncertainty about the functioning of the system. For example, a change in the capacity of a facility due to failure is related to reliability. The disaster may occur due to capacity restrictions or the closure of a number of facilities. Berman et al. described reliability issues for hospitals in Toronto. They located the facilities when the infliction of damage to facilities is likely. They also pointed to the closure of some hospitals because of the SARS outbreak and demanding urban emergency services and facilities mentioned in other locations [15].

Snyder considered two developed P-median models related to reliability, while Schneider formulated and solved a variety of developed locating models considering reliability [14]. Torres et al. developed a two-objective model to minimize the distance between health centers to emergency centers and also the costs where they select a number of treatment centers from them, which provides 24-hour emergency services. They also presented a harmony meta-heuristic algorithm to solve their model. They also presented an approach to reduce the search space to increase the efficiency of their introduction [16].

Hoseinpour and Ahmadi-Javid considered the design of an immobile service system in which each facility’s service process was subject to the risk of interruptions. They maximized the difference
between the service provider’s profit and the sum of customers’ transportation and waiting costs. To solve large-sized instances an efficient Lagrangian-based solution algorithm was developed [17]. Shishebori and Yousefi provided an efficient mixed integer linear programming model for a robust and reliable medical service (MS) center location network design problem (RR/MSL/NDP), which simultaneously tacked uncertain parameters, system disruptions, and investment budget constraint into account [18].

2.2. Hierarchical facility location problem (HFLP)
A hierarchical system refers to a system in which the facilities are unilaterally interrelated from top-down or bottom-up at different levels of service. The level of service (consisting of various facilities) is created when the lowest level is Level 1 and the top level is Level k. Some typical applications of HFLP modeling include healthcare systems, emergency medical services, education systems, production–distribution networks, and telecommunication systems. Sahin & Sural classified hierarchical models based on flow patterns, objectives functions, service type, and space configuration [19].

- Flow pattern describes how products/services flow between network nodes. In a network with a single-flow model, demand is formed at zero levels and the priority of facilities levels, taking into account the type of service, and ends at the highest level (or vice versa). Also, in a multi-flow network, the demand may be produced at any level such as \( k (k = 1 \ldots m) \) production and responded at higher (or lower) levels [20].

- Service availability: Every system, according to the availability of a variety of services, is classified into successively inclusive and successively exclusive systems at any level of the hierarchy [21].

In a successively inclusive system, each facility at a higher level presents the services provided at the lower-level facility, together with a range of different services to customers (e.g., healthcare system), but in a successively exclusive system, at each level of the hierarchy, facilities provide special services to customers and distinct from other levels.

- Spatial configuration: If any subset of a large set is considered as a hierarchy, the relationship between these subsets is formed into coherent and incoherent types. In a coherent system, all application areas allocated to an especial facility are assigned with an identical facility at a higher facility, while the incoherent systems have no restriction on the spatial configuration of levels. In the following, medical care models will be discussed among hierarchical location models. The Calvo and Marks’s model is among the first hierarchical location models for health centers. In this continuous multi-stream model, k levels are considered for facilities (with limited capacity) and the number of facilities is determined by the model [21]. Narula and Ogbu presented a two-level model taking into account the limitation of capacity for the facilities, so that Level 1 is related to health centers and Level 2 is related to hospitals, and 5 heuristic methods are proposed to solve the model and report the results of some calculations [22]. Parr provided a successively inclusive, multi-flow incoherent model. He briefly pointed out a set of guidelines to make a more realistic model [23]. Tien and El-Tell solved the model presented by Kahlo and Marx. It was actually a local successively inclusive model. Then, they presented a successively inclusive model, which actually was the extended version of Kalu and Marx’s model, and began to solve their model [24]. Gerrard and Church [25] provided a two-level model, and Boffey et al. [26] presented a three-level model concerning the possibility of transferring patients from treatment centers at lower levels to
treatment centers at higher levels. Galvão et al. also offered a three-level model that, in addition to assuming the transfer of patients to higher levels, added capacity restrictions of facilities at high levels and solved the problem with the help of Lagrangian Relaxation (LR) [27]. Yasenovskiy and Hodgson offered a three-level model. The model was based on the assumption that, in reality, people may not always go to the nearest facility, and for some reasons, they may prefer facilities at higher levels that provide better services but in the farthest distance to those at a lower level with limited services but in the nearer distance [28]. Hodgson and Jacobs developed the previous model based on the patient’s behavior considering different possibilities to transfer the patient to a higher-level relief center according to the required service level for the patient [29]. Considering the uncertainty in the field of health locating is important for two main reasons:

Pouraliakbarimamaghani et al. proposed a location-allocation model for a capacitated healthcare system. They developed a discrete modeling framework to determine the optimal number of facilities among candidates and optimal allocations of existing customers for operating health centers in a coverage distance so that the total sum of customer and operating facility costs are minimized. The setup costs of the hospitals were based on the costs of customers, fixed costs of establishing healthcare centers and costs based on the available resources at each level of hospitals. In this paper, the idea of a hierarchical structure was used. Two levels of service were considered in hospitals including low and high levels and sections at different levels that provide different types of services. To solve the model, they proposed two meta-heuristic algorithms, including genetic algorithm, simulated annealing and their combination [30].

1. Decisions taken for locating are long-term decisions and the possibility of making a change in the decisions after the locating process is very costly.
2. These facilities play an important role in the disaster, and lack of attention to the uncertainty of disasters leads to inefficient solutions.

2.3. Facility location models in disasters

Since the extent and intensity of natural disasters are on the rise due to factors such as population growth, climate change and global connectivity, it is predicted that the current aids are insufficient [31]. On the other hand, the nature of natural disasters is in such a way that responding to them should be done in a short time. In such emergency and sophisticated conditions, the decision-maker must respond quickly and effectively to problems and transfer the wounded from damaged areas to designated centers. For locating relief facilities, the first study was conducted in 1971 by Toregas et al. They raised this issue in the envelope form, and then used linear programming methods to solve it [32]. In the studies in this area, Chang et al. modeled locating facilities and distributed relief supplies to flood relief, according to different flood scenarios, using a random two-step plan, taking into account the uncertainty of demand for the relief [33]. Najafi et al. proposed a randomized multi-objective, multi-product, multi-course model and a model with few types of vehicle to distribute relief tools and transport the wounded after the earthquake and the optimization under uncertainty. To ensure the proper functioning of the distribution program after the earthquake, we presented a robust method [34]. Bozorgi et al. provided a robust planning approach for the design of relief logistics services in the conditions of uncertainty. They considered the parameters of supply, demand and cost of production and transportation of relief goods at the same time as the inaccurate parameters of the problem. This was the first time that these three factors of uncertainty were considered at the same time for robust modeling [35]. Shen et al. formulated the problem of locating facilities in critical conditions and solved the model using a
number of heuristic methods. They assumed that when a facility fails, the customers’ demands will be assigned to other facilities [36]. Shavandi and Mahlooji developed hierarchical location-allocation models for congested systems, for example, in health care systems, by employing a queueing theory in a fuzzy framework. The parameters of models were approximately evaluated and stated as fuzzy numbers. The coverage of demand nodes was also considered approximately and was stated by the degree of membership. Using queueing theory and fuzzy conditions, both referral and nested hierarchical models were developed for the Location Set Covering Problem (LSCP) [37]. Motallebi Nasrabadi et al. considered a problem consisting of both patients’ and service providers' requirements (i.e., accessibility vs. costs) for locating healthcare facilities, allocating service units to those facilities, and determining the facilities’ capacities. They captured both short-term and long-term uncertainties at the modeling stage. The queuing theory was incorporated to consider stochastic demand and service time as a short-term uncertainty, and a service level measurement. They also demonstrated a way in which a linearized model can become more efficient by eliminating excessive binary variables when service level constraints are approximated using their properties. Additionally, the long-term demographic variations were captured through robust optimization to develop a robust model. To solve the problem under investigation, an evolutionary solution method was designed and its performance was investigated under different settings [38]. Mestre et al. proposed two location-allocation models for handling uncertainty in the strategic planning of hospital networks. The purpose of these models was to be able to reorganize the hospital network system when the decision-maker seeks to improve local access while minimizing costs. The key features in the design of hospital networks, including the hospitals that provide multiple services in a hierarchical structure, were modeled on a planning horizon in which network changes may occur. These models include various assumptions about decisions that should be made without complete knowledge of the uncertain parameters, and also decisions that should be made after uncertainty was identified. The demand uncertainties were modeled through a set of discrete scenarios. Both models applied to the case study of the Portuguese National Health Service [39]. Ghezavati et al. proposed a hierarchical location model in the disaster relief chain under uncertainty to determine the timing of customer service. They considered the possibility of closing the roadways for relief operations in a disaster. In the surveyed network, a higher-level relief center offers all the services provided by lower-relief facilities. A robust optimization method and chance-constrained programming were used [40]. As an option to control the different types of uncertainty, the fuzzy set theory can be used. For example, Canós et al [41], Darzentas [42], Rao and Saraswati [43] all addressed the problem of fuzzy locating. However, all these models assumed the parameters of the problem to be definite. On the other hand, scholars like Zhou and Liu [44], considering the demand as a fuzzy case, have investigated the problem of locating facilities and assigning demand points to them according to the capacity restrictions of facilities. in Table 1 a summary of the features of the reviewed articles is presented.
The innovations presented in this paper include: provide a robust model to locate hierarchy health care facilities in disaster conditions, take into account the possibility of establishing temporary relief centers and transferring medical service teams from permanent medical centers to temporary ones as well as the possibility of transferring patients from lower level centers to medical cares that are at higher levels, provide two proposed meta-heuristic methods based on harmony search algorithm, and Tabu search hybrid algorithm with variable neighborhood search algorithm, and finally provide lower bound based the Lagrangian relaxation method for proposed problem.
3. Robust optimization approach
In this section, the robust optimization is described.

3.1. Robust optimization method based on scenario
Mulvey et al. in 1995 proposed two important definitions that involve two types of robustness: solution robustness and model robustness [45]:

In an optimization model, a solution is robust if the model remains nearly optimal in all scenarios, and a model is robust when the solution is nearly feasible in all scenarios. Mullvey et al. presented a robust optimization model considering cost-benefit analysis concerning solution robustness and model robustness. In the robust optimization model, there are two types of variables: control variables and design variables. Design variables are decided before understanding the possible parameters and cannot change after understanding the possible parameters. Control variables are moderated after a certain understanding of uncertainty parameters. The robust optimization model provided by Malloy et al. is formulated as follows:

Initially, a number of symbols associated with the model are introduced. Consider the following linear programming model that contains random parameters:

\[
\begin{align*}
\text{Min } & \ f(x,y)=cx+dy \\
\text{S.t.} & \ Ax =b \\
& \ Bx +Cy =e \\
& \ x, y \geq 0
\end{align*}
\]

\(x\) is the vector of design variables and \(y\) is the vector of control variables. \(A\), \(B\) and \(C\) are coefficients of parameters and \(b\) and \(e\) are vectors of parameters (Right hand values). \(A\) and \(b\) are definite values, while \(B\), \(C\) and \(e\) have uncertainties. A specific understanding of uncertainty parameter is called scenario to which symbols are allocated and its possibility is specified by \(\rho\) to show a set of scenarios \(\Omega\) used. The coefficients of uncertainty are allocated as \(e_s, B_s\) and \(C_s\) for each scenario \(s \in \Omega\). Besides, since control variable \(y\) is moderated after understanding the scenario, \(y_s\) can be allocated to scenario \(s\). Due to the uncertainty of parameters, the model is likely to be unjustified for a number of scenarios. Therefore, \(\eta_s\) represents the infeasibility of the model under scenario \(s\). If the model is feasible, \(\eta_s\) is equal to zero; otherwise, it will receive a positive value from the following equations. The Mullvey et al.'s model is formulated as follows:

\[
\begin{align*}
\text{Min } & \ \sigma(x,y_1, y_2, ..., y_s) + \gamma \rho(\eta_1, \eta_s, ..., \eta_s) \\
\text{S.t.} & \ Ax =b \\
& \ B_s x +C_s y_s +\eta_s =e_s \\
x_s, y_s, \eta_s \geq 0, \forall s \in \Omega
\end{align*}
\]

There are two terms in the objective function; the first one indicates the solution robustness, and the second shows the model robustness by weight \(\gamma\). Then, the two terms are discussed. High
variance \( f_s = (x, y_s) \) indicates that decision has a high risk. Otherwise, a small change in parameters with uncertainty can lead to a great change in the value of the measuring function. Malloy et al. used this term to show solution robustness. \( \delta \) is the weight allocated to solution variance.

\[
\text{Min } Z = \sum_{s \in \Omega} \rho_s f_s + \delta \sum_{s \in \Omega} \rho_s \left( f_s - \sum_{s \in \Omega} \rho_s f_s \right)^2, \sum_s \rho_s = 1
\]  

As you can see, a second-order term exists in the above formula.

Yu and Li suggested that the formula proposed by Mullvey et al. needs complex and numerous calculations due to non-linearity. They instead offered the following formulation [46]:

\[
\text{Min } Z = \sum_{s \in \Omega} \rho_s F_s + \delta \sum_{s \in \Omega} \rho_s \left( F_s - \sum_{s \in \Omega} \rho_s F_s \right) + 2\theta_s \]  

\[
F_s - \sum_{s \in \Omega} \rho_s F_s + \theta_s \geq 0 \quad \forall s \in \Omega
\]  

\[
\theta_s \geq 0 \quad s \in \Omega
\]

where \( \theta_s \) is linearizing variable and Equations (11) and (12) are used for making the variance term linear.

In addition to the above, in Mullvey's model objective function, we can add another term indicating penalty for non-compliance of some model constraints to some scenarios.

\[
\text{Min } Z = \sum_{s \in \Omega} \rho_s F_s + \delta \sum_{s \in \Omega} \rho_s \left( F_s - \sum_{s \in \Omega} \rho_s F_s \right) + 2\theta_s + \gamma \sum_{s \in \Omega} \rho_s \eta_s
\]  

In Equation (13), \( \gamma \) is the weight considered for the violation of the model constraints and shows the cost-benefit analysis between the model and solution robustness. Moreover, the violation of the model from the restrictions in scenario \( s \) is indicated by \( \eta_s \).

4. Problem statement

Healthcare facilities are the most important city facilities that are directly involved in providing the health of individuals and society. Locating healthcare facilities are among the important strategies to improve the performance of these facilities and reduce the delay in a normal situation. Moreover, during disasters, these centers are important, because one of the most important measures that need to be taken in the time of disasters is the optimal allocation of affected people to these centers. Therefore, considering the appropriate location of such facilities in normal and critical conditions can simultaneously increase the level of satisfaction with the facilities and reduce losses. In the disaster conditions, one of the most important issues is how to deal with the wounded and provide them with health services in the shortest time possible. One of the actions that have a significant impact on the willingness of governments to the disaster is the proper
planning during the establishment of relief and treatment centers. The most important issue in this area is to choose a location for medical centers. These facilities should provide proper services to people before the disaster and be available too. On the other hand, during the disaster, these centers are expected to have a good performance. Choosing the wrong place for this kind of center, on the one hand, can cause downtime and out-of-service conditions during disasters in the case of proximity to the center of disasters. On the other hand, if it is too far away from the center of the disaster, the transfer of patients and mortality rate increase.

During the disaster, establishing outpatient centers can help classify the wounded in terms of injury level, and the patients with minor injuries can be treated in the relief center, and only the highly injured ones will be transferred to permanent centers.

If establishing temporary relief centers is ignored, the wounded will be transferred to permanent centers without prioritizing, and due to limited resources, the transfer of casualties increases.

On the other hand, due to the transfer of patients with low injury rates to permanent treatment centers, congestion and chaos occur in the centers. Also, due to the lack of accommodation camps together with permanent treatment centers, patients are encountered with low levels of damage after treatment and will be confused in this regard (as to what to do). On the other hand, during establishing temporary relief centers, paying attention to the capacity of human resources services is also important. Thus, sending equipment and medical teams to temporary centers should be considered.

**Assumptions**

The assumptions of a robust model are as follows:
- In this model, \( K \) types of facility and \( K \) levels of service are considered.
- In \( k \)-type health centers, services are provided at the service levels of \( c=1, 2, ...K \), and service levels of \( c=k+1, ..., K \) are not provided in these centers.
- Each of the demand points can be a candidate point to be considered for the establishment of a facility.
- It is assumed that the higher facilities cover all facility services of the lower level; in other words, the model is successively inclusive.
- In each node, you can build at least one facility, and the capacity for each facility is limited for any specific service.
- If demand for each service of a facility is more than the capacity, a certain amount of penalty is considered.
- In this model, the service reception of any level can begin; in other words, the model has a multi-flow pattern.
- The possibility of transferring patients from any level to a higher level exists if needed.
- In other words, the model has referral property.
- The cost of establishing all the facilities of a kind is equal.
- It is assumed that the facilities of the same type have equal capacity in each service level.
- It is assumed in the first scenario that normal conditions prevail and no disaster occurred, and in other scenarios that an event happened in different regions with different intensities.
- At the time of disasters, in addition to the permanent hierarchical facilities, the temporary facilities to be established are assumed, if necessary, which can only provide low-level services.
- Permanent facilities can transfer part of the low-level facilities to the temporary facilities, if necessary.
- In the case of failure, permanent facilities are not able to provide any level of services, but they can send basic equipment and medical teams to the temporary facilities.

Collections

$cu$: Set of demand points (candidate), $cu = \{1, 2, ..., n\}$ and $i$, $j$, $h$ are corresponding indices.

$K_s$: Levels of medical facilities and provided services, $k = 1, 2, ..., K$ and $k, c$ are the corresponding indices.

$S$: Set of provided scenarios, $s = 1, 2, ..., S$

Parameters:

$Ra_{c \rightarrow c+l}$: The transition rate from service level $c$ to the service level $c + l$

$Q_{kc}$: The capacity of facility type $k$ at service level $c$

$u_{ic}$: Percentage of demanded service type $c$, at demand point $i$ in scenario $s$

$dam_{js}$: Failure probability of facility $j$ in scenario $s$

$w_{is}$: Demand of node $i$ in scenario $s$

$Bud_1$: The total budget available for permanent facilities (Currency).

$Bud_2$: The total budget available for temporary facilities in each scenario (Currency).

$cap$: The capacity of temporary facility

$CO_k$: The cost of establishing permanent facility

$Ct$: The cost of establishing temporary facility

$d_{ijs}$: The distance between demand point $i$ to facility $j$ in scenario $s$

$d_{jhs}$: The distance between facility $j$ and $h$ in Scenario $s$

$\rho_s$: The probability of occurrence of scenario $s$

$\lambda$: The weight allocated to solution variance

$M$: A large number

Decision variables

$x_{ijcs}$: Demand (number of patients) in point $i$ with level of service demanded equal to $c$ from a facility located in point $j$ at level $k$ and scenario $s$

$r_{jkhcs}$: Population referred from facility $j$ to facility $h$ of type $k$ to provide service level $c$ in scenario $s$

$eq_{jkhcs}$: Amount of low-level service that is transferred from facility $j$ of type $k$ to temporary facilities in scenario $s$

$y_{jk}$: Equal to 1 if a facility type $k$ to be established in point $j$; otherwise, it is 0

$temp_{js}$: Equal to 1 if a temporary facility in scenario $s$ to be established in point $j$; otherwise, it is 0.

$sh_{jcs}$: Shortage cost related to lack of capacity of facility $j$ at service level $c$ in scenario $s$

The robust mathematical model is expressed as follows:
Minimize \[ Z = \sum_{j=1}^{n} \sum_{k=1}^{K} CO_j \times y_{jk} + \sum_{j=1}^{n} \sum_{k=1}^{K} \sum_{l=1}^{K} d_{jk \times l} \times x_{jk\times l} \] 
\[ + \sum_{j=1}^{n} Ct_{\text{temp}_j} + \sum_{j=1}^{n} \sum_{l=1}^{K} \sum_{c=1}^{K} d_{jk \times l} \times p_{jkl} + \sum_{j=1}^{n} \sum_{k=1}^{K} d_{jk \times eq_{jkl}} \]
\[ + \sum_{j=1}^{n} \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{c=1}^{K} d_{jk \times l} \times x_{jk\times l} \times \sum_{l=1}^{K} \sum_{c=1}^{K} d_{jk \times l} \times p_{jkl} + \sum_{j=1}^{n} \sum_{k=1}^{K} d_{jk \times eq_{jkl}} \]
\[ + \sum_{j=1}^{n} \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{c=1}^{K} d_{jk \times l} \times eq_{jkl} \] 

\[ (\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{K} \sum_{l=1}^{K} d_{jk \times l} \times x_{jk\times l} + \sum_{j=1}^{n} Ct_{\text{temp}_j} + \sum_{j=1}^{n} \sum_{l=1}^{K} \sum_{c=1}^{K} d_{jk \times l} \times p_{jkl} + \sum_{j=1}^{n} \sum_{k=1}^{K} d_{jk \times eq_{jkl}} ) \]
\[ + \lambda \sum_{i=1}^{S} \sum_{j=1}^{n} \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{c=1}^{K} d_{jk \times l} \times eq_{jkl} + 2\theta_i \]
\[ + \gamma \sum_{j=1}^{n} \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{c=1}^{K} \rho_{j\times k\times l} \]

Subject to:
\[ \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{K-1} \sum_{l=1}^{K} \sum_{c=1}^{S} x_{ijk\times c} = 0 \]  
\[ (15) \]
\[ \sum_{j=1}^{n} \sum_{k=1}^{K-1} \sum_{l=1}^{K} \sum_{c=1}^{S} r_{j\times k\times l\times c} = 0 \]  
\[ (16) \]
\[ \sum_{k=1}^{K} \sum_{i=1}^{K} x_{ijk\times c} = u_i \times w_{is} \quad c = 1, 2, 3, ..., K \quad i \in \text{cu} \quad s \in S \]  
\[ (17) \]
\[ \sum_{j=1}^{n} \text{temp}_{j} = 0 \]  
\[ (18) \]
\[ \sum_{k=1}^{K} y_{jk} \times \text{dam}_{js} + \sum_{j=1}^{n} \sum_{s=1}^{S} \text{temp}_{js} \leq 1 \quad j \in \text{cu} \]  
\[ (19) \]
\[ \sum_{i=1}^{n} x_{ijk\times c} \leq M \times y_{jk} \times \text{dam}_{js} \quad j \in \text{cu} \quad k=2,3,..,K \quad s \in S \]  
\[ (20) \]
\[ \sum_{i=1}^{n} x_{ijk\times c} \leq M \times (y_{jk} \times \text{dam}_{js} + \text{temp}_{js}) \quad j \in \text{cu}, \quad k=1 \quad s \in S \]  
\[ (21) \]
\[ \sum_{j=1}^{n} \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{c=1}^{K} \sum_{d=1}^{K} r_{j\times k\times l\times c} = \sum_{j=1}^{n} Ra_{(c+d)} \times x_{ijk\times l} \quad j \in \text{cu} \quad c=1,2,3,..,K-1, \quad c+l \in Ks, \quad s \in S \]  
\[ (22) \]
\begin{align*}
\sum_{j=1}^{n} r_{jkcs} & \leq M \times y_{hk} \times dam_{hs} \quad h \in cu, j \neq h, \quad k, c \in Ks, s \in S \\
(23) \\
\sum_{h=1}^{n} eq_{jkhs} & \leq Q_{kc} \times y_{jk} \quad j \neq h, \quad j \in cu, \quad k=1,2,\ldots,K, \quad s \in S, c=1 \\
(24) \\
\sum_{j=1}^{n} eq_{jkhs} & \leq \beta \times cap \times temp_{hs} \quad j \neq h, \quad h \in cu, \quad k=Ks, \quad s \in S \\
(25) \\
\sum_{i=1}^{n} \sum_{h=1}^{n} \sum_{k=1}^{K} x_{ijkcs} + \sum_{h=1}^{n} \sum_{k=1}^{K} r_{jkhcs} + \sum_{h=1}^{n} \sum_{k=1}^{K} eq_{jkhs} - \sum_{i=1}^{n} \sum_{k=1}^{K} Ra_{c(l+1)} x_{ijkcs} \\
- \sum_{k=1}^{K} Q_{kc} \times y_{jk} \times dam_{hs} - (cap \times temp_{js} + \sum_{h=1}^{n} \sum_{k=1}^{K} eq_{shjs}) = Sh_{jcs} \\
\quad j \in cu, \quad s \in S, \quad c \in Ks, \quad c + l \in Ks \\
(26) \\
\sum_{j=1}^{n} \sum_{k=1}^{3} CO_{k} \times y_{jk} & \leq Bud 1 \\
(27) \\
\sum_{j=1}^{n} Ct \times temp_{js} & \leq Bud 2, \quad s \in S \\
(28) \\
(\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{K} d_{ij} \times x_{ijkcs} + \sum_{j=1}^{n} \sum_{k=1}^{K} d_{jk} r_{jkhcs} + \sum_{j=1}^{n} \sum_{k=1}^{K} d_{jk} eq_{jkhs}) \\
- \sum_{s'}^{s} \rho_{s'} (\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{K} d_{ij} \times x_{ijkcs}') \\
+ \sum_{j=1}^{n} \sum_{h=1}^{n} \sum_{k=1}^{C} d_{jk} r_{jkhcs}' + \sum_{j=1}^{n} \sum_{h=1}^{n} \sum_{k=1}^{K} d_{jk} eq_{jkhs}') + \theta_{i} \geq 0, \quad s \in S \\
(29) \\
x_{ijkcs}, eq_{jkhs}, Sh_{jcs}, r_{jkhcs} \geq 0 \quad i, j, h \in cu, \quad c, k \in Ks, s \in S \\
(30) \\
y_{jk}, temp_{js} \in \{0,1\}, \quad j \in cu, \quad k \in Ks, s \in S \\
(31)
\end{align*}

The objective function of the model is based on the Malloy's robust model objective function, and the description of other parts is as before. Equations (15) and (16) ensure that no facility can
provide service above its level. Equation (17) ensures that all demands in each demand point i are assigned to the facilities. Equation (18) states that in the case of normal situations, there is no temporary facility. Equation (19) indicates that at any point, at most a permanent or temporary facility can be found (However, in any scenario, subject to failure of permanent facility in one area, a temporary facility can be found too). Equations (20) and (21) state that the service can be received at any particular level only if a facility of the same level or a higher level (non-exclusive) is available in this point. Equation (22) states the number of people transferred to the service at the higher level according to the defined rate. In fact, this service cannot meet the requirements for this group of people based on the type of their need. Equation (23) guarantees that the patients are transferred wherever a facility is established. Equation (24) expresses permanent facilities, if established, can send equipment from Service Level 1 (low-level) to temporary facilities. Equation (25) expresses that the temporary facilities, if established, can receive up to β% of their capacity the facilities from a higher facility (permanent facility). Equation (26) calculates the capacity shortage for all services provided by each center. For this purpose, the number of people who directly refer for their service to the center in addition to patients who have been transferred to the specific service at the center plus the equipment transferred from the desired permanent center to other temporary centers minus the number of people who are referred to other services and minus the capacity of the center in the desired service offers the shortage of the center. As can be seen, for each service, if the number of referred patients is more than capacity, the corresponding penalty will be considered in the objective function. Equations (27) and (28) express the budget restrictions and Equation (29) is the linearization constraint in the Malloy's model. Equations (30) and (31) define the type of decision variables.

5. Solution methods
Locating facilities is a complex and difficult problem [47]. Therefore, to solve the problem in higher orders, two meta-heuristic methods and a lower-bound method are proposed, which are described in this section.

5.1. Harmony Search Algorithm
In recent years, with the development of computers, meta-heuristic methods have received a lot of attention. Among efficient meta-heuristic algorithms, one can point out to harmony search algorithm, inspired by the method of making a piece of music [48]. Due to applicability to discrete and continuous optimization problems, simple concept, low parameters, and easy implementation, this algorithm is one of the most common optimization algorithms in recent years [49]. Any musician or musical instrument shows a decision variable in this method. During the algorithm, each musician plays a note and, in fact, each decision variable is allocated a value. The objective of algorithm iteration is to find the best harmony between the musicians or the global optimal point. The steps of harmony search algorithm can be expressed as follows:

Step 1: Set algorithm parameters
In this algorithm, like any meta-heuristic algorithm, the algorithm parameters need to be set. (Note that our problem space is discrete and, therefore, we explain the algorithm in the discrete space.) The harmony search algorithm parameters include the number of vectors in harmony memory size (HMS), probability of selecting harmony memory consideration rate (HMCRR), probability of setting and changing pitch adjustment rate (PAR) and band width distance (BW) that is used for the problems in continuous spaces.
And finally, the maximum number of algorithm iterations is indicated as MaxIt. In this algorithm, each solution is called harmony and represented by an N-component vector. A harmony memory (HM) matrix is built using several solutions or harmonies.

\[
\begin{bmatrix}
  x_1^1 & x_2^1 & \cdots & x_N^1 & f(x^1) \\
  x_1^2 & x_2^2 & \cdots & x_N^2 & f(x^2) \\
  \vdots & \vdots & \ddots & \vdots & \vdots \\
  x_1^{HMS} & x_2^{HMS} & \cdots & x_N^{HMS} & f(x^{HMS})
\end{bmatrix}
\]

(32)

**Step 2: Create first generation (initial values) for algorithm**

In this step, as many as HMS of the first generation, harmony vectors are created randomly and stored in Harmony Memory (HM). Then, the objective function is calculated and stored for each of these vectors and stored in Fig. 1 matrix.

**Step 3: Create a new harmony**

Each harmony vector includes problem variables. To create value for \(i\)-th variable, we first produce a random number between zero and one. Then, we compare this number with parameter HMCR. If it was less, a value is chosen for \(i\)-th variable from the memory matrix and \(i\)-th column. Otherwise, a random value is chosen from the search space for \(i\)-th variable with the probability as \((1 - \text{HMCR})\). If a value was chosen from the memory matrix, we can set it to probability PAR.

**Step 4: Update harmony memory**

The value of newly generated harmony is compared with the worst harmony available in the matrix memory. If it is better than the worst harmony available in matrix memory, the old harmony is replaced by the new one.

**Step 5: Check stopping condition**

Steps 3 and 4 continue until a certain number of iterations (MaxIt), and the best solution is reported as the solution to the problem [50]. Despite the suitability of harmony search algorithm, this algorithm quickly converges in some cases. Also, this algorithm with a greedy nature seeks to improve the worst solution available in the harmony memory, which reduces the variety of solutions on harmony memory and leads to being trapped by the local optimum [51].

Given the problems listed for harmony search algorithm, improving and increasing the efficiency of the changes required to develop these algorithms are presented: Among the modifications of the harmony search algorithm is to improve memory harmony, so that a cumulative idea was used for producing harmony memory. The cumulative idea was first proposed in 2008 by Degertekin. It acts in a way that instead of producing all memory members randomly, it produces random solutions twice the HMS and places as many as HMS of the best solutions in harmony memory [52]. Among other improvements in the fixed algorithm is to avoid considering the choice probability parameter from harmony memory HMCR so that the value of this parameter over the consequent iterations of algorithm changes linearly. This change occurs as follows [51]:

16
\[
HMCR(t) = HMCR_f + \frac{(HMCR_f - HMCR_i)(t - 1)}{MaxIt - 1}
\]

\(t \in \{1, ..., MaxIt\}\)  

where \(HMCR_i\) represents \(HMCR\) value in the first iteration and \(HMCR_f\) represents this value in the last iteration. The harmony search algorithm's pseudo-code is shown as follows:

**Input:** initialize parameters and harmony memory  
**Repeat**  
\(t = 1\)  
**While** (not_termination)  
{  
**for** (\(I = 1\) to \(n\))  
  **if** (\(\text{rand1} \leq \text{HMCR}(t)\))  
    \(x(i)\) will be randomly chosen from harmony memory  
  **if** (\(\text{rand2} \leq \text{PAR}\)) pitch adjustment  
    \(x(i) = x(i \pm 1)\)  
  **end if**  
  else  
    \(x(i) = \text{random selection}\)  
  **end if**  
**end for**  
\(t = t + 1\)  
evaluate the fitness of each vector  
update harmony memory  
update \(\text{HMCR}(t)\)  
}  

5.2. Solution string  
The model presented in this study is a location-allocation model. After conducting examinations, it was found out that if the type of established facility is known in each location, the solution time is dramatically reduced and the model is solved easily and quickly using the exact model. Therefore, to solve the model, the integration of the exact method and meta-heuristic method was used. In each iteration, using the proposed meta-heuristic algorithm, the type of treatment centers located in any place is added to the model as an input parameter given to the model. Then, the model is solved using CPLEX solver and returns the resulting solution. To display the solution, the two-part string was used so that the first part has a length of \(n\), and for the establishment of permanent facilities, as the second part has multiple subparts equal to scenarios minus 1 each of length \(n\), and shows how to establish temporary facilities at any candidate point in any scenario. It is worth noting that in scenario 1, no temporary center can be established. In the first string, in each cell of the corresponding string, numbers 0-\(k\) can be set. Number 0 means that in the corresponding candidate point, no facilities were built. Numbers 1-\(k\) represents the facility establishment types 1-\(k\) in relevant places. In Figure 1, an example is presented with 10 candidate points and three kinds of permanent facilities considering three scenarios.
According to Figure 1, in candidate points 1, 6 and 7, no facilities were established. In Points 2 and 9, facilities type 1 were established, and in Points 5 and 8, facilities type 2 were established, and finally, in Points 3, 4 and 10, facilities Type 3 were established. The second part of the string, as shown in Figure 1, includes subparts with length 10, each related to one scenario except the first scenario. In each cell of the corresponding subparts, numbers 0 and 1 can be put. Number 0 means that in the desired candidate point, no temporary facility is built, and number 1 means establishing a temporary facility in the candidate point. In Figure 2, in the first scenario (in normal conditions), no temporary facility is established. Therefore, scenario 1 is not presented in the solution string. In scenario 2, in the above example, in each of points 1, 2, 4, 8, 9 and 10, a temporary facility is established. In scenario 3 in points 3, 5, 6, 7 and 9, a temporary facility is established. It should be noted that the presented string is interpreted in such a manner that if a permanent facility exists in node $i$ and if this facility does not fail in one scenario, no temporary facility is established in this place without paying attention to the part related to the corresponding scenario.

5.3. Tabu search algorithm

Tabu search algorithm is a meta-heuristic optimization algorithm that was first introduced in 1986 by Glover [53]. This method is widely used as a method of optimization. This technique is a holistic approach to conduct research for achieving good results in a complex solution space. To achieve optimum results in an optimization problem, the tabu search algorithm begins to move from the initial solution and the list of prohibited actions applies. This list contains previous changes in the solution string and makes it possible for those previous changes to remain unchanged at least in the next several moves. Then, the algorithm selects the best neighbor solution from the current neighbors. If the solution is not on the tabu list, the algorithm moves to solution neighbors; otherwise, the algorithm checks a measure called the aspiration criterion. According to the aspiration criterion, if the neighbor solution is better than the best neighbor result found, the algorithm will move to this neighborhood, even if the solution is on the tabu list. After the algorithm moved to the neighboring result, the tabu list is updated, which means that the previous change that led to the current result is located in the tabu list to prevent the algorithm from returning and creating a cycle. In fact, a tabu list is a tool in the tabu search algorithm by which the algorithm is prevented from being trapped in the local optima. After placing the previous move in tabu list, some movements that were previously located in the list are removed from the list. The time the parameters are located on the list is determined by a parameter called tabu list length. Moving from the present solution to the neighbor solution continues until the stopping condition occurs. Different stopping conditions can be considered for the algorithm. The parameters used in this section are presented in Table 2.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sol</td>
<td>Current answer</td>
</tr>
<tr>
<td>$S'$</td>
<td>The answer resulting from neighborhood</td>
</tr>
<tr>
<td>$Sol_{best}$</td>
<td>The best found answer</td>
</tr>
</tbody>
</table>
The number of iterations that if the neighborhood does not reach the best answer, the algorithm goes on to the next neighborhood.

MaxIt Maximum repetition of algorithm

StCon The maximum number of repetitions, that if the best solution is not changed, the algorithm stops.

del Counter of the number of not reaching improvement (in the neighborhood)

No Im p Counter of the number of times that the best answer is not improved. (Total algorithm)

it Counter of algorithm iterations

5.3.1. Proposed hybrid tabu search algorithm

To produce an initial solution to the proposed hybrid tabu search algorithm, a simplified mathematical model was used. In this model, only the demand level of each service anywhere in the scenarios and the cost of facilities established at different levels and the shortage cost in each scenario, regardless of the cost allocated to patients, number and location of permanent facilities and also temporary facilities will be determined in each scenario. In the proposed tabu search algorithm, a variable neighborhood search algorithm is used. Its main idea is to change the neighborhood using local search. A variable neighborhood search algorithm was first proposed in 1997 by Mladenović & Hansen [54]. Simplicity in implementation and quality of results obtained from Variable Neighborhood Search (VNS) quickly made this method a good way to solve optimization problems. Assume that $A_l$ for $l = \{1, 2, \ldots, l_{\text{max}}\}$ is a predetermined neighborhood structure and $A_l(x)$ is the set of neighborhoods of $x$ under the structure of $A_l$. Algorithm VNS has two main phases, shaking and local search. In the shaking phase, the algorithm moves to a neighborhood solution ($S'$) from the current solution using $i$-th neighborhood structure. Besides, the local search phase is searched on $S'$ using neighborhood search methods to reach optimized $S'$. Now, in the moving or non-moving part, if the obtained local optima are better than the current solution ($Sol$), it will be replaced by it. Otherwise, the next neighborhood structure $A_{l+1}$ is used to continue searching. This search continues until $l<l_{\text{max}}$. The variable neighborhood algorithm's pseudo-code is expressed as follows [55]:

\textbf{Input:} a set of neighborhood structures $A_l$, $l=1,2,\ldots,l_{\text{max}}$

$Sol=$generate initial solution();

\textbf{Repeat}

$l=1$;

\textbf{While}($l \leq l_{\text{max}}$)

\{

$s' = \text{shaking (sol, } A_l)$

$s'' = \text{Localsearch (s')}$

\textbf{if} $f(s'') \leq f(sol)$

$Sol \leftarrow s''$

$l = l + 1$;

\textbf{else}

\}

\textbf{End While}
\[l = l + 1;\]

\(\text{Until stopping condition are met;}\)

Output: The best solution;

**Initial solution generation phase**

According to the variable neighborhood search algorithm, the initial solution should be local optimal [55]. Accordingly, in each iteration, the latest result obtained from the tabu search algorithm up to this iteration is given to this algorithm.

**Neighborhood creation phase (shaking phase)**

The purpose of this phase is to create a sudden change in solution [55]. Each of the methods for the creation of the neighborhood is called a \(k\)-neighborhood method. In the proposed algorithm, based on the studies conducted based on the impact on the performance of the algorithm, eight neighborhoods (operations) were considered. The eight neighborhoods are listed in Table 3.

\{Please insert table 3 about here.\}

**Local search phase**

In the shaking phase, if the initial results taken from the tabu search algorithm improve, they will be given to the tabu search algorithm to apply the local search using alternative techniques. In this phase, the algorithm in each iteration creates some neighborhoods with interchange heuristic introduced by Narula & Ogbe [22] (which is actually the developed form of the heuristic method presented by Teitz & Bart [56]). Using this technique in each iteration, a \(t\)-type facility \((t=1,2,3,...)\) in position \(I\) may be replaced by zero in position \(j\) (zero value means there is no facility in the corresponding point) or \(t\)-type facility in position \(I\) is likely to be replaced by a facility with other types in position \(j\). After each neighborhood was produced, if the result is better than the current solution, the resulting solution is replaced by the current solution, and the neighborhood led to improvement is transferred to tabu list and the list is updated, and the counter of the number of times of not improved solution becomes zero in the neighborhood (del); otherwise, one unit is added to the counter.

If the counter reaches a particular value given as the algorithm input (StCon), the production of this neighborhood is stopped and the obtained result is considered as the local optimum solution. Finally, the obtained result returns to the VNS algorithm again and this round trip continues until the algorithm stops. The stopping condition in the proposed tabu search algorithm is in two forms and each condition occurred first will stop the algorithm. One of the conditions is no improvement of the solution in a particular number of consequent iteration \((Nolmp > \text{StCon})\) and the other condition is the maximum number of iterations \((it > \text{MaxIt})\).

5.4. **Setting parameters using Taguchi method**

The parameters of meta-heuristic algorithms affect their performance. A suitable combination of these factors can greatly improve the performance of the algorithms. There are some ways to design tests. One of the first ways that were presented in this area is the factorial method in which the number of tests is obtained from \(N = L^m\)

A major drawback of this approach is that if there are too many variables, too many tests are needed and it is not appropriate in terms of time and cost. The taguchi method is a widely used method of setting parameters [57]. Therefore, in this paper, the Taguchi method is used to set
parameters. In the tabu search algorithm, four variables as controllable factors are determined as follows:
The total number of iterations of the algorithm in which no improvement occurred ($StCon_{it}$), number of not improved solutions in local searches ($StCon$), total number of iterations of the algorithm ($MaxIt$), and tabu list length ($D$).
For Taguchi testing, three levels were considered and the $L_9$ Taguchi design was used. Based on the results of experiments obtained from the Taguchi test, the values for each of the desired parameters mentioned above are equal to 100, 10, 30 and finally, the tabu list length was considered equal to the square root of the number of neighborhoods divided by two. In the harmony search algorithm, five variables were determined as controllable factors, including harmony memory size ($HMS$), rate of choice from harmony memory in the first iteration ($HMCR_I$), rate of choice from harmony memory in the last iteration ($HMCR_F$), pace adjustment rate ($PAR$), and number of algorithm iterations ($MaxIt$). According to the results obtained from the Taguchi test, the considered values for the above parameters are 1000, 0.2, 0.5, 0.9, and 300, respectively.

5.5. Lower bound of Lagrangian relaxation
To compare the performance of algorithms in a large-size problem, the Lagrangian relaxation method is used to produce a lower bound, and in this section, the explanation will be discussed.

5.5.1. Lagrangian relaxation method
Lagrangian relaxation method is one of the most effective and efficient ways in a discrete optimization problem which can be used to produce lower bounds in minimization problems [58, 59]. In a given problem, the choice of constraint or constraints is important and those constraints should be relaxed that have much impact on the complexity of the problem. In a minimization problem, a more optimal value of Lagrangian dual function means stronger relaxation (release) [60]. Consider the following problem:

$$ Z = \text{Min} \ cx $$
$$ Ax \geq b $$
$$ Dx \geq e $$
$$ x \geq 0 $$

With the Lagrangian relaxation method, the problem is converted into the following form:

$$ Z = \text{Min} \ cx + \lambda(b-Ax) $$
$$ Dx \geq e $$
$$ x \geq 0 $$

With the addition of a constraint to the objective function, a negative value is added to the objective function ($\lambda \geq 0$), and thus the solution to the second question as a lower bound on the main problem can be raised. However, with the removal of constraint 35, the problem solution does not get worse. Thus, the solution to the second question without constraint 35 is a lower bound. In a Lagrangian maximization problem, as it finds a lower bound, we look for the highest objective function value (the greatest lower bound).
5.5.2. Applying Lagrangian relaxation method to proposed model

In the Lagrangian problem, selecting the constraint that is added to the objective function is very important. In fact, a constraint should be selected to have both a great impact on the reduction of solution time (complexity) of the problem and elimination of the constraint and adding it to objective function should lead to the creation of good lower bounds (in minimization problem) or good upper bounds in a maximization problem. After conducting investigations and selecting different constraints for the Lagrangian relaxation of constraints (20), (21), and (23), the model was presented in Section 1.4 and the objective function of the Lagrangian relaxation method took the following shape:

\[
\min \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{K} \sum_{c=1}^{C} \sum_{s=1}^{S} \lambda_{ijkcs} (x_{ijkcs} - M \times y_{jk} \times dam_{js} + M \times temp_{js}) \\
+ \sum_{h=1}^{n} \sum_{k=1}^{K} \sum_{c=1}^{C} \sum_{s=1}^{S} \beta_{hkcs} (\sum_{j=1}^{n} r_{jhc} - M \times y_{hk} \times dam_{hs})
\]

Other constraints of the Lagrangian optimization approach are the same as the robust model presented in Section 1.4. The Lagrangian optimization approach determining appropriate values for the Lagrangian coefficients is also important. A general method to determine the optimum values used for the Lagrangian coefficients is the sub-gradient optimization algorithm [22, 61]. To find the optimal values for the Lagrangian coefficients, the sub-gradient optimization algorithm applied in this study [62]. In this algorithm, for the stopping condition, the maximum number of 100 iterations or a step size less than or equal to 0.01 is used. The initial values of Lagrangian coefficients were considered as zero. The step size was initially determined as much as two. If no improvement is achieved at the lower bound after 10 iterations, this value is divided by two. The pseudo code for the sub-gradient algorithm is expressed as follows [61]:

1. **Initialization**
   1.1. Find a feasible solution \( Z_h \) using the tabu search algorithm described in Section 5-3;
   1.2. Initialize Lagrangian multipliers;
   1.3. Initialize \( L_b := -\infty \); lower bound for \( LR \);
   1.4. Initialize parameters and counters;
2. **Do While** stop conditions not met:
   2.1. Solve Lagrangian problem \( LR \) obtaining \( Z_{LR} \);
   2.2. Update best lower bound so far, \( L_b := \max [L_b; Z_{LR}] \);
   2.3. Compute sub-gradient vectors;
   2.4. Update step size;
   2.5. Update Lagrangian multipliers;
   2.6. Check whether stop condition met;
3. **END**

To calculate the size of the step in each iteration of the sub-gradient algorithm, a possible solution is required. The possible required solution is obtained using a hybrid tabu search algorithm described in Section 1.3.5. This algorithm produces a feasible solution with a high performance which remains unchanged in the entire algorithm.
6. Computational results

6.1. Analyzing difficulty of problem and production of test problems

To produce sample problems, the difficulty of the problem and its dependence on the parameters were examined. For this reason, 12 problems with eight candidate points, 12 problems with ten candidate points and 12 problems with 12 candidate points in different budgets were produced randomly in a two-dimensional space [10*10]. Then, the patients' demands and the shortage penalty were set so that in case of the availability of sufficient budget in all parts, the candidate facility should be established. To compare the solution time, the demand value in each of the 12 problems with similar candidate points was considered as fixed. The maximum total time for the problems was considered 9000 seconds and the problems were solved using CPLEX. Figure 2 shows the average time needed to solve problems with different levels of budget based on the number of established candidates compared to all candidate points.

According to Figure 2, when the number of established candidates is low relative to the total candidate points (in low budgets), the complexity of problems is also low and the problem-solving time is low. With the increased budget and increasing the number of facilities that can be established, the complexity of problems increases and the solution time goes up. This trend continues so long as we reach high budgets, as in most of the candidate points, facilities are established. Afterward, the complexity of the problems and the time to solve problems are reduced too. According to Figure 2, it seems that the complexity is mostly related to when budget and demand are so that the need to establish facilities varies between 55% and 75% of candidate points. To produce sample problems, this result was used and the random samples were produced in both difficult and easy samples. To produce sample problems in small dimensions, the coordinates of points were determined randomly in a two-dimensional space [10*10], and in the problems with large dimensions, the coordinates of each of points were randomly determined in a two-dimensional space [50*50]. Also, in sample problems, three service centers and three levels were considered. To produce scenarios in sample problems, 4 different events were considered. Thus, in the first case, it was assumed that no disaster occurred and normalcy prevails. In all the next three modes, a point was randomly generated in the solution space and was considered as the center of the disaster. In this case, a destruction radius was considered. Table 4 shows the destruction radius to handle large and small problems for different aspects. In sample problems, it was assumed that all facilities within the destruction radius will fail with a probability of 80%.

To randomly calculate the demand rate for each of the initial points, the population resident in each of the demand points were generated in the range [30,200]. The demand rate in each of the points was considered proportional to the resident population (equal to 30% of the population). To calculate the demand rate in the disaster, Index $if_{k}$ was defined. This index is the incidence rate in $k$-th disaster-inflicted region. The index value for the points that have no distance from the incidence center is equal to 80%, and for the points that are far from the disaster center, it is twice the radius. This index decreases linearly from 80% to 10%. Index $if_{k}$ was considered as 10% in normalcy (when no disaster was occurred).
In this research, the rate of demand from services was obtained at different levels based on the parameters estimated by Oppong through studying the data collected from medical centers located in Suhum, Ghana. This demanded service was considered as follows [63]:

\[ u_c = (u_1, u_2, u_3) = (0.609, 0.203, 0.188) \]  \hspace{1cm} (42)

Also, for the production of sample problems, it was assumed that 20% of the injured are directly transferred to the Level 1 service, 43% are referred to the Level 2 service, and 10% are referred to Level 3 service. The transfer rate of the injured from Level 2 service to Level 3 service is considered as 25%. Ultimately, for producing sample problems, it was assumed that every temporary facility established in critical situations can receive equipment from permanent centers and medical staff (Level 1 service) up to 20% of its maximum capacity. Finally, for the production of sample problems, it was assumed, in terms of disaster, that any temporary facility established can receive up to 20% of its capacity the permanent centers, equipment, and medical personnel (service level).

6.2. Computational results
In this subsection, the results of the proposed algorithms are analyzed for the small and large-size problem. The results of the exact method, lower bound and two meta-heuristics in small-size problems are shown in Tables 5 and 6. In the small problems, the gap is obtained by the deviation percentage of the objective function resulting from the proposed algorithms and the objective function obtained by the exact method. This value is calculated by Equation (43).

\[ \text{GAP} = \frac{Z - Z_{\text{best}}}{Z} \times 100 \]  \hspace{1cm} (43)

In this equation, \( Z_{\text{best}} \) is the solution obtained from CPLEX and \( Z \) is the solution obtained from the algorithm. In the following tables, \( CP \), \( LR \), \( HS \), and \( C-TS \) indicate CPLEX, Lagrangian lower bound, harmony search algorithm and hybrid tabu search algorithm methods, respectively.

{Please insert table 5 about here.}

As shown in Tables 5 and 6, the average error of solution methods is not much different. The average error of the hybrid Tabu search algorithm in the small simple and difficult problems is equal to 0.06% and 0.27%. The average gap of the harmony search algorithm for these two problems is 4.79% and 4.26%, respectively. In both categories of problems, the Lagrangian method offered a lower bound with 23.96% and 20.3% gaps. According to the results of the hybrid Tabu search algorithm and harmony search algorithm as an upper bound and the result of the relaxation method as a lower bound, on average, a 20% range for the optimum solution is provided. In terms of solution time in the category of difficult problems, the solution time significantly increased compared to those obtained from a simple problem. In simple problems, on average, the amount of time needed to solve these problems in CPLEX, harmony search algorithm and hybrid tabu search algorithm was about 504 seconds, 282 seconds and 145 seconds, respectively. However, in difficult problems, on average, solution time is about 979 seconds, 570 seconds and 176 seconds. In Figure 3, the solution time of algorithms is indicated for simple small problems.
6.3. Results of large-scale computing
The results obtained from the proposed algorithms in large-size problems with simple and difficult problems are shown in Tables 7 and 8.

As indicated in Tables 7 and 8, the average gaps of the hybrid Tabu search algorithm and harmony algorithm related to the lower bound are about 26% and 27%. Since the Lagrangian relaxation method in small-size problems showed a 22% gap compared to the optimum solution and assuming that the quality of Lagrangian relaxation method in large-size problem does not decline, as an optimistic assumption, we can conclude that the average error of hybrid tabu search algorithm and harmony algorithm is at most about 4% and 5%, respectively, in the large-size problems. In the simple problems, compared to the difficult problems, the average time to solve is better for both meta-heuristic algorithms according to Tables 6 and 7. In simple and difficult problems, the solution time in hybrid tabu search algorithm is, on average, about 4361 seconds and 4915 seconds, and in harmony search algorithm, it is equal to 2536.86 and 2272.15 seconds, respectively. It seems that less solution time in harmony search algorithm compared to hybrid tabu search algorithm on a large scale is the result of the rapid convergence of this algorithm.

6.3. Impact of robust solution on results
To evaluate the performance of the robust model, an example concerning different scenarios was analyzed. In this example, six scenarios were considered. Initially, a crisp model under normal conditions (the first scenario) and a robust model were implemented. Then, taking into account the decisions of the first stage resulting from the two models that included the type and location for the establishment of permanent health facilities, a crisp model with data from each of the six scenarios were solved. The results are shown in figure 4.

As shown in Figure 4, the objective function value obtained from the crisp model under normal conditions in different scenarios has changed a lot. The crisp model acts better than the robust model only in normal scenarios, and in the event of a disaster (Scenarios 2-6), the robust model
has better performance. In this example, the average value of the objective function in different scenarios for the robust and crisp models is equal to 4755.808 and 3456.132. The standard deviation of the objective function in the robust model is equal to 103.905 and in the crisp model is equal to 1106.52.

7. Conclusion
In this paper, a mathematical robust model was presented for the construction of permanent hierarchical healthcare centers and temporary relief centers in normal and critical situations. Then, to solve the problem, two meta-heuristic methods, harmony search and hybrid tabu search and variable search algorithm were proposed. Moreover, a lower bound-based Lagrangian relaxation method was presented to obtain a lower bound for the proposed problem. To check the quality of the proposed algorithms, we evaluated the impact of parameters on the difficulty of the problem, and given the amount of available budget and the ratio of facilities that can be established in all the candidate points, the sample problems were divided into two categories: difficult and simple. Then, in each category, some problems were generated in large and small dimensions. In the category of small and simple problems, the quality of solutions resulting from the meta-heuristic algorithms was appropriate. In this way, compared to the exact solution of the hybrid tabu search and harmony algorithm, they had an average error as much as 4.48% and 0.18%, respectively. The distance of the lower bound resulting from the Lagrangian relaxation algorithm was about 22% compared to the optimum solution. On large scales, the results of the algorithms were compared to the lower bound of the Lagrangian method. The mean deviation of the solution of the hybrid tabu search algorithm and harmony algorithm compared to the proposed lower bound by the Lagrangian relaxation method is, on average, equal to 27% and 26%. According to the 22% error of the lower bound resulting from Lagrangian relaxation algorithm in small sizes, compared to the optimum solution, and assuming that the quality of Lagrangian relaxation algorithm in the great problems is not worst, it can be concluded that the average error of hybrid tabu search and harmony algorithm on large scales is at most about 5% and 6%, respectively. Finally, the reduction in costs in the case of using a robust model in disaster in a numerical example was investigated. The suggestion made for future studies is to consider the criteria such as patient's waiting time for service, and increasing equity in offering healthcare services.

References


---

Dr. Mahdi Alinaghian is an Associate Professor at Isfahan university of technology. He received PhD in Industrial Engineering, from Iran university of technology. His research interests include Discrete Event simulation, uncertain programming, supply chain management, Meta heuristic Algorithms. His published research articles appear in, EXPERT SYST APPL, NSPEC, J INTELL MANUF, IJAMT, and etc.

Department of Industrial and Systems Engineering, Isfahan University of Technology, 84156-83111 Isfahan, Iran

---

1-1-2
Dr. Seyed Reza Hejazi is a Full Professor at Isfahan University of Technology. He received PhD in Industrial Engineering, from Tarbiat Modares University. His research interests include Fuzzy systems, Operations Research, multiple criteria decision making, Supply Chain Management and etc. His published research articles appear in, International Journal of Production Research, Fuzzy sets and systems, Computers & Operations Research and etc.

Department of Industrial and Systems Engineering, Isfahan University of Technology, 84156-83111 Isfahan, Iran

Ms. Nooshin Bajol earned a master’s degree in industrial engineering from Isfahan University of Technology. Her research interests include Discrete Event simulation, uncertain programming, supply chain management, Meta heuristic Algorithms.

Department of Industrial and Systems Engineering, Isfahan University of Technology, 84156-83111 Isfahan, Iran
Dr. Kosar Sadeghi Velni earned a bachelor's in industrial engineering from Tehran University and a Master's degree from Isfahan university of technology. She received PhD in Industrial Engineering from Tehran university. her research interests include, supply chain management, Meta heuristic Algorithms and etc.

Department of Industrial Engineering, Fouman Faculty of Engineering, University of Tehran, 013-34915103 Fouman, Iran.

Figure and table captions

Table 1: Summary of reviewed articles
Figure 1: Example of a solution string for the problem with k = 3 and 10 candidate points
Table 2: The tabu search algorithm parameters
Table 3: The neighborhoods in tabu search algorithm
Figure 2: The average solution time according to the number of established facilities compared to all candidate points
Table 4: Failure radius intended to problems in different dimensions
Table 5: Results of small size simple problems solving (CP: CPLEX, LR: Lagrangian lower bound, HS: harmony search algorithm, C-TS: hybrid tabu search algorithm)
Table 6: Results of small size difficult problems solving (CP: CPLEX, LR: Lagrangian lower bound, HS: harmony search algorithm, C-TS: hybrid tabu search algorithm)
Figure 3: comparing the solution time of CPLEX and HS and TS algorithms for small simple problems solving
Table 7: Results of large size simple problems solving with proposed algorithms (LR: Lagrangian lower bound, HS: harmony search algorithm, C-TS: hybrid tabu search algorithm)
Table 8: Results of large size difficult problems solving with proposed algorithms (LR: Lagrangian lower bound, HS: harmony search algorithm, C-TS: hybrid tabu search algorithm)
Figure 4: The objective function value obtained from robust model, and crisp model, for all scenarios
<table>
<thead>
<tr>
<th>Table 1: Summary of reviewed articles</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>hierarchical facilities</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Eaton et al.[5]</td>
</tr>
<tr>
<td>Jacobs et al.[6]</td>
</tr>
<tr>
<td>McAleer and Naqvi [7]</td>
</tr>
<tr>
<td>Osinuga et al.[8]</td>
</tr>
<tr>
<td>Cardoso et al.[9]</td>
</tr>
<tr>
<td>Dogan et al. [10]</td>
</tr>
<tr>
<td>Daskin et al. [11]</td>
</tr>
<tr>
<td>Carson and Batta [12]</td>
</tr>
<tr>
<td>Revelle et al. [13]</td>
</tr>
<tr>
<td>Bermanet al.[15]</td>
</tr>
<tr>
<td>Snyder [16]</td>
</tr>
<tr>
<td>Torres et al. [17]</td>
</tr>
<tr>
<td>Hoseinpour and Ahmadi-Javid [18]</td>
</tr>
<tr>
<td>Shishebori and Yousefi [19]</td>
</tr>
<tr>
<td>Calvo and Marks [22]</td>
</tr>
<tr>
<td>Narula and Ogdu [23]</td>
</tr>
<tr>
<td>Parr [24]</td>
</tr>
<tr>
<td>Authors</td>
</tr>
<tr>
<td>----------------------------------</td>
</tr>
<tr>
<td>Tien and El-Tell [25]</td>
</tr>
<tr>
<td>Gerrard and Church [26]</td>
</tr>
<tr>
<td>Boffey et al. [27]</td>
</tr>
<tr>
<td>Galvão et al. [28]</td>
</tr>
<tr>
<td>Yasenovskyi and Hodgson [29]</td>
</tr>
<tr>
<td>Hodgson and Jacobs [30]</td>
</tr>
<tr>
<td>Pouraliakbarimamaghani et al. [31]</td>
</tr>
<tr>
<td>Toregas et al. [33]</td>
</tr>
<tr>
<td>Chang et al. [34]</td>
</tr>
<tr>
<td>Najafi et al. [35]</td>
</tr>
<tr>
<td>Bozorgi et al. [36]</td>
</tr>
<tr>
<td>Shen et al. [37]</td>
</tr>
<tr>
<td>Shavandi and Mahlooji [38]</td>
</tr>
<tr>
<td>Motalebi Nasrabadí et al. [39]</td>
</tr>
<tr>
<td>Mestre et al. [40]</td>
</tr>
<tr>
<td>Ghezavti et al. [41]</td>
</tr>
<tr>
<td>This study</td>
</tr>
</tbody>
</table>
Table 2: The tabu search algorithm parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sol$</td>
<td>Current answer</td>
</tr>
<tr>
<td>$S'$</td>
<td>The answer resulting from neighborhood</td>
</tr>
<tr>
<td>$Sol_{best}$</td>
<td>The best found answer</td>
</tr>
<tr>
<td>$StCon$</td>
<td>The number of iterations that if the neighborhood does not reach the best answer, the algorithm goes on to the next neighborhood.</td>
</tr>
<tr>
<td>$MaxIt$</td>
<td>Maximum repetition of algorithm</td>
</tr>
<tr>
<td>$StCon_{it}$</td>
<td>The maximum number of repetitions, that if the best solution is not changed, the algorithm stops.</td>
</tr>
<tr>
<td>$del$</td>
<td>Counter of the number of not reaching improvement (in the neighborhood)</td>
</tr>
<tr>
<td>$No Imp$</td>
<td>Counter of the number of times that the best answer is not improved. (Total algorithm)</td>
</tr>
<tr>
<td>$it$</td>
<td>Counter of algorithm iterations</td>
</tr>
</tbody>
</table>

Table 3: The neighborhoods in tabu search algorithm

<table>
<thead>
<tr>
<th>How to create neighborhood</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
</tbody>
</table>

Table 4: Failure radius intended to problems in different dimensions

<table>
<thead>
<tr>
<th>The radius of failure in the small-scale problems</th>
<th>The radius of failure in the macro-scale issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
</tr>
</tbody>
</table>
Table 5: Results of small size simple problems solving (CP: CPLEX, LR: Lagrangian lower bound, HS: harmony search algorithm, C-TS: hybrid tabu search algorithm)

<table>
<thead>
<tr>
<th>Problem</th>
<th>cu</th>
<th>CP</th>
<th>LR</th>
<th>HS</th>
<th>C-TS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Z</td>
<td>T</td>
<td>Z</td>
<td>Gap%</td>
</tr>
<tr>
<td>P1e</td>
<td>7</td>
<td>992.31</td>
<td>1.3</td>
<td>586.71</td>
<td>28.78</td>
</tr>
<tr>
<td>P2e</td>
<td>7</td>
<td>743.28</td>
<td>8.6</td>
<td>554.63</td>
<td>17.31</td>
</tr>
<tr>
<td>P3e</td>
<td>8</td>
<td>1093.01</td>
<td>8.22</td>
<td>688.04</td>
<td>26.98</td>
</tr>
<tr>
<td>P4e</td>
<td>8</td>
<td>926.32</td>
<td>12.07</td>
<td>672.56</td>
<td>16.59</td>
</tr>
<tr>
<td>P5e</td>
<td>9</td>
<td>1256.83</td>
<td>20.52</td>
<td>785.06</td>
<td>25.60</td>
</tr>
<tr>
<td>P6e</td>
<td>9</td>
<td>1022.77</td>
<td>24.38</td>
<td>757.29</td>
<td>19.11</td>
</tr>
<tr>
<td>P7e</td>
<td>10</td>
<td>1595.56</td>
<td>50.96</td>
<td>978.30</td>
<td>19.88</td>
</tr>
<tr>
<td>P8e</td>
<td>10</td>
<td>1275.32</td>
<td>31.84</td>
<td>928.82</td>
<td>17.76</td>
</tr>
<tr>
<td>P9e</td>
<td>11</td>
<td>1873.78</td>
<td>58.64</td>
<td>1134.56</td>
<td>17.56</td>
</tr>
<tr>
<td>P10e</td>
<td>11</td>
<td>1458.15</td>
<td>255.11</td>
<td>1064.98</td>
<td>18.04</td>
</tr>
<tr>
<td>P11e</td>
<td>12</td>
<td>2134.72</td>
<td>533.96</td>
<td>1280.02</td>
<td>17.08</td>
</tr>
<tr>
<td>P12e</td>
<td>12</td>
<td>2005.25</td>
<td>2483.81</td>
<td>1244.66</td>
<td>16.98</td>
</tr>
<tr>
<td>P13e</td>
<td>12</td>
<td>1655.25</td>
<td>3068.68</td>
<td>1195.11</td>
<td>18.73</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>1387.12</td>
<td>504.46</td>
<td>1115.44</td>
<td>20.03</td>
</tr>
</tbody>
</table>

Table 6: Results of small size difficult problems solving (CP: CPLEX, LR: Lagrangian lower bound, HS: harmony search algorithm, C-TS: hybrid tabu search algorithm)

<table>
<thead>
<tr>
<th>Problem</th>
<th>Cu</th>
<th>CP</th>
<th>LR</th>
<th>HS</th>
<th>C-TS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Z</td>
<td>T</td>
<td>Z</td>
<td>Gap%</td>
</tr>
<tr>
<td>P1d</td>
<td>7</td>
<td>806.96</td>
<td>8.16</td>
<td>634.63</td>
<td>21.35</td>
</tr>
<tr>
<td>P2d</td>
<td>7</td>
<td>777.64</td>
<td>8.64</td>
<td>554.63</td>
<td>28.47</td>
</tr>
<tr>
<td>P3d</td>
<td>7</td>
<td>757.88</td>
<td>9.07</td>
<td>554.63</td>
<td>26.81</td>
</tr>
<tr>
<td>P4d</td>
<td>8</td>
<td>985.98</td>
<td>17.43</td>
<td>772.56</td>
<td>21.64</td>
</tr>
<tr>
<td>P5d</td>
<td>8</td>
<td>962.68</td>
<td>27.68</td>
<td>762.56</td>
<td>20.78</td>
</tr>
<tr>
<td>P6d</td>
<td>9</td>
<td>1149.84</td>
<td>28.48</td>
<td>867.29</td>
<td>24.57</td>
</tr>
<tr>
<td>P7d</td>
<td>9</td>
<td>1094.93</td>
<td>74.90</td>
<td>867.29</td>
<td>20.79</td>
</tr>
<tr>
<td>P8d</td>
<td>10</td>
<td>1408.03</td>
<td>162.36</td>
<td>1068.82</td>
<td>24.09</td>
</tr>
<tr>
<td>P9d</td>
<td>10</td>
<td>1372.26</td>
<td>714.83</td>
<td>1068.82</td>
<td>22.11</td>
</tr>
<tr>
<td>P10d</td>
<td>11</td>
<td>1592.65</td>
<td>1121.13</td>
<td>1214.98</td>
<td>23.73</td>
</tr>
<tr>
<td>P11d</td>
<td>12</td>
<td>1975.62</td>
<td>1610.63</td>
<td>1444.66</td>
<td>26.87</td>
</tr>
<tr>
<td>P12d</td>
<td>12</td>
<td>1908.49</td>
<td>2765.41</td>
<td>1411.26</td>
<td>26.05</td>
</tr>
<tr>
<td>P13d</td>
<td>12</td>
<td>1822.95</td>
<td>6180.75</td>
<td>1385.11</td>
<td>24.01</td>
</tr>
<tr>
<td>average</td>
<td></td>
<td>1278.14</td>
<td>979.19</td>
<td>969.78</td>
<td>23.96</td>
</tr>
</tbody>
</table>
### Table 7: Results of large size simple problems solving with proposed algorithms (LR: Lagrangian lower bound, HS: harmony search algorithm, C-TS: hybrid tabu search algorithm)

<table>
<thead>
<tr>
<th>Problem</th>
<th>LR cu</th>
<th>Z</th>
<th>HS cu</th>
<th>Z</th>
<th>T</th>
<th>GAP%</th>
<th>C-TS cu</th>
<th>Z</th>
<th>T</th>
<th>GAP%</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1be</td>
<td>15</td>
<td>1603.46</td>
<td></td>
<td>2032.12</td>
<td>918.33</td>
<td>18.97</td>
<td></td>
<td>1991.03</td>
<td>702.29</td>
<td>19.47</td>
</tr>
<tr>
<td>P2be</td>
<td>20</td>
<td>1859.01</td>
<td></td>
<td>2451.01</td>
<td>1367.34</td>
<td>21.17</td>
<td></td>
<td>2416.19</td>
<td>1035.14</td>
<td>23.07</td>
</tr>
<tr>
<td>P3be</td>
<td>25</td>
<td>1776.51</td>
<td></td>
<td>2392.10</td>
<td>1511.55</td>
<td>26.57</td>
<td></td>
<td>2373.56</td>
<td>2254.35</td>
<td>25.14</td>
</tr>
<tr>
<td>P4be</td>
<td>30</td>
<td>2229.43</td>
<td></td>
<td>3147.89</td>
<td>1979.45</td>
<td>30.95</td>
<td></td>
<td>3093.25</td>
<td>2732.03</td>
<td>28.97</td>
</tr>
<tr>
<td>P5be</td>
<td>35</td>
<td>3072.74</td>
<td></td>
<td>4405.79</td>
<td>2266.44</td>
<td>30.10</td>
<td></td>
<td>4309.81</td>
<td>4097.15</td>
<td>29.65</td>
</tr>
<tr>
<td>P6be</td>
<td>40</td>
<td>5368.19</td>
<td></td>
<td>7534.28</td>
<td>2761.48</td>
<td>29.12</td>
<td></td>
<td>7519.56</td>
<td>6728.64</td>
<td>28.63</td>
</tr>
<tr>
<td>P7be</td>
<td>50</td>
<td>6549.19</td>
<td></td>
<td>10432.62</td>
<td>3205.86</td>
<td>31.59</td>
<td></td>
<td>9244.90</td>
<td>7360.12</td>
<td>30.17</td>
</tr>
<tr>
<td>P8be</td>
<td>60</td>
<td>5644.17</td>
<td></td>
<td>10963.70</td>
<td>4166.78</td>
<td>40.26</td>
<td></td>
<td>10019.43</td>
<td>8133.42</td>
<td>34.64</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>1919.81</td>
<td></td>
<td>5419.93</td>
<td>2272.15</td>
<td>28.59</td>
<td></td>
<td>5120.96</td>
<td>4361.11</td>
<td>27.34</td>
</tr>
</tbody>
</table>

### Table 8: Results of large size difficult problems solving with proposed algorithms (LR: Lagrangian lower bound, HS: harmony search algorithm, C-TS: hybrid tabu search algorithm)

<table>
<thead>
<tr>
<th>Problem</th>
<th>LR cu</th>
<th>Z</th>
<th>HS cu</th>
<th>Z</th>
<th>T</th>
<th>GAP%</th>
<th>C-TS cu</th>
<th>Z</th>
<th>T</th>
<th>GAP%</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1bd</td>
<td>15</td>
<td>1437.17</td>
<td></td>
<td>1770.35</td>
<td>954.27</td>
<td>17.89</td>
<td></td>
<td>1733.35</td>
<td>763.28</td>
<td>17.08</td>
</tr>
<tr>
<td>P2bd</td>
<td>20</td>
<td>1519.58</td>
<td></td>
<td>1998.99</td>
<td>1466.95</td>
<td>23.98</td>
<td></td>
<td>1974.08</td>
<td>1235.61</td>
<td>23.02</td>
</tr>
<tr>
<td>P3bd</td>
<td>25</td>
<td>1841.93</td>
<td></td>
<td>2396.04</td>
<td>1662.69</td>
<td>23.12</td>
<td></td>
<td>2381.65</td>
<td>2961.56</td>
<td>22.66</td>
</tr>
<tr>
<td>P4bd</td>
<td>30</td>
<td>2303.24</td>
<td></td>
<td>3148.40</td>
<td>2162.66</td>
<td>26.84</td>
<td></td>
<td>3087.02</td>
<td>4110.11</td>
<td>25.38</td>
</tr>
<tr>
<td>P5bd</td>
<td>35</td>
<td>2426.39</td>
<td></td>
<td>3403.72</td>
<td>2947.30</td>
<td>28.71</td>
<td></td>
<td>3343.71</td>
<td>5324.23</td>
<td>27.43</td>
</tr>
<tr>
<td>P6bd</td>
<td>45</td>
<td>4888.73</td>
<td></td>
<td>6905.43</td>
<td>3140.64</td>
<td>29.20</td>
<td></td>
<td>6884.12</td>
<td>7028.03</td>
<td>28.98</td>
</tr>
<tr>
<td>P7bd</td>
<td>50</td>
<td>4729.15</td>
<td></td>
<td>7079.68</td>
<td>3608.92</td>
<td>33.20</td>
<td></td>
<td>6863.74</td>
<td>7899.14</td>
<td>31.09</td>
</tr>
<tr>
<td>P8bd</td>
<td>60</td>
<td>4937.89</td>
<td></td>
<td>7081.47</td>
<td>4351.46</td>
<td>30.27</td>
<td></td>
<td>6981.09</td>
<td>10000.00</td>
<td>29.26</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>3010.51</td>
<td></td>
<td>4220.51</td>
<td>2536.86</td>
<td>26.65</td>
<td></td>
<td>4156.09</td>
<td>4915.24</td>
<td>25.61</td>
</tr>
</tbody>
</table>

Figure 1: Example of a solution string for the problem with k = 3 and 10 candidate points

<table>
<thead>
<tr>
<th>0 1 3 3 2 0 0 2 1 3</th>
<th>1 1 0 1 0 0 1 1 1 1</th>
<th>0 0 1 0 1 1 1 0 1 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>First part</td>
<td>Second part</td>
<td></td>
</tr>
</tbody>
</table>

36
Figure 2: The average solution time according to the number of established facilities compared to all candidate points.

Figure 3: Comparing the solution time of CPLEX and HS and TS algorithms for small simple problems solving.
Figure 4: The objective function value obtained from robust model, and crisp model, for all scenarios