Journal of Uncertain Systems
Vol.5, No.3, pp.202-226, 2011
Online at: www.jus.org.uk

A Fuzzy Multi-objective Programming for Optimization of Hierarchical Service Centre Locations

Ali Pahlavani*, Mohammad Saidi Mehrabad
Department of Industrial Engineering, Iran University of Science and Technology, Narmak, Tehran, 16844, Iran

Received 19 November 2009; Revised 21 August 2010

Abstract

This paper presents a model for locating hierarchical service centres with a nested nature. Customers' demands for different services are assumed to be in four levels. We utilize fuzzy theory to deal with the uncertain nature of cost and travelling time. For this purpose, travel times and costs are denoted as triangular fuzzy numbers. The model is formulated as a special version of fuzzy goal programming in which the objectives are satisfaction grades of the original objectives as minimization of average travel time, maximization of demand coverage and minimization of total costs. Two methods are proposed to obtain the satisfaction grades of travel time and cost objectives. Satisfaction grade of demand coverage objective is measured by comparing the really established facilities with the appropriate number of facilities in terms of before-defined distance measure for each level of facilities. This prevents demand undercover or over cover. Cost function involves establishment cost of facilities including fixed and variable costs which both of them are dependent on the hierarchy level. To solve the problem, Tabu Search and Simulated Annealing, two well-known meta-heuristics are employed. A set of experiments are performed to show the efficiency of the algorithms.

Keywords: location problem, hierarchical service centre, multi-objective optimization, fuzzy goal programming, satisfaction grade

1 Introduction

In recent years, location planning issues in service sector have attracted some researchers. Meanwhile, the planning of outlets for emergency services such as medical systems, police and fire departments has a distinguished position due to its higher impact on public safety. Service systems could be classified into two main categories according to their service providing style: mobile or service-to-customer in contrast to immobile or customer-to-service. This paper considers those services which are provided by immobile facilities (server, outlet and service centre is also possible). Obvious examples are medical systems and police offices. An important aspect of this type of services is its facilities' hierarchical structure. The literature of facility location problem is extensively involved by various location models for single-level systems (i.e. a single facility type) [26].

In hierarchical service systems, facilities at different levels provide different types of services. However, there is often a linkage between the different levels, which makes the problem not separable. They can be classified according to their structure as nested and non-nested systems [24]. In a nested system, the high-level facilities provide low-level services too, while in non-nested systems, each level offers its own special service. A hierarchical system is labelled as coherent if all customers of a particular low-level facility are the customers of a particular high-level facility as well. In a referral system, the users can go to a higher-level facility only when referred by a low-level facility. A non-referral system lacks such restriction [21].

The wide applications of hierarchical systems and their vital importance to human life have motivated us to present a multi-objective model for finding the optimal locations for the facilities of a service sector organization with hierarchical structure. The problem of locating service centres involves multiple objectives usually in conflict. Along with the cost minimization which is a common objective used in many location studies, distance or travelling time minimization is the other important objective that reflects the accessibility of service systems and affects customers' mind in patronizing a service centre. In the case of vital services, facilities must be located in such a way that the customers in demand points have access to the service within a reasonable distance or time. Furthermore, the

* Corresponding author. Email: pahlavani@iust.ac.ir (A. Pahlavani); Tel:+982127871057.

adequacy of available service capacity could be mentioned as an objective, i.e., the capacity should not be under or over a necessary range.

The objectives considered in the model are minimization of average travelling times for facilities of each level, minimization of total costs related to facilities establishment and maximization of adequacy of demand coverage. So, the number of objectives is dependent on the number of hierarchy levels. The modelling approach is very close to the concept of Goal Programming (GP). In fact, it maximizes satisfaction grade of the decision maker about objective function value instead of minimizing deviation of each objective from its goal.

Certainly, some aspects of the problem are uncertain. For example, time to reach the facilities is uncertain due to some factors such as varying vehicles to get location, path traffic variation in different periods and so on. Similarly the establishment cost of facilities is uncertain due to estimation errors and oscillation of prices. Fuzzy numbers and linguistic values are utilized to deal with uncertainties. The model also allows the decision maker (DM) to express his/her preference to the partially achievement of the goals because it is not possible to construct a network of facilities in which all of our goals are fully satisfied, i.e., some of the goals may not be achieved or partially achieved. A modified version of fuzzy goal programming approach is engaged to incorporate the decision makers' imprecise aspiration levels for the goals.

Since the developed multi-objective service centre location model is highly non-linear, it cannot be solved using ordinary optimization methods and we have to utilize approximation algorithms. For this purpose, we use Simulated Annealing and Tabu Search, two well-know local search meta-heuristic algorithms. They are tested and compared on a large set of randomly generated instances.

The rest of the paper is as follows; a brief survey on the related literature is provided in Section 2. A theoretical background is given in Section 3. Section 4 explains the developed fuzzy multi-objective model. The solution algorithms are described in Section 5. Computational results of experiments are given in Section 6 and finally Section 7 concludes the paper and gives some issues for future research.

2 Literature Survey

All classic models in the location science such as p-median [15] and Maximal Covering Location Problem [9] are single-objective. In spite of a rich literature existing on location theory, comparatively small emphasis was put on analyzing multi-objective models. However, in real-world location decisions, a variety of objectives could be considered. Especially public service systems must meet a variety of objectives in location and allocation decisions. The need for the multi-objective framework to plan public facilities has been discussed by some authors [3].

A number of multi-objective formulations and objectives to be considered in location problems are described by Current et al. [10]. ReVelle [25] extended maximal covering location problem in the case of two-objective. Similarly Heller et al. [16] discussed the use of a multiple objective p-median model for locating emergency medical service facilities.

Meanwhile works dedicated to multi-objective modelling of hierarchical facility location problems are scarce ([13, 6]).

Multi-objective location models can be solved using mathematical approaches such as GP originally proposed by Charnes et al. [7]. GP as a powerful multi-objective decision-making approach is analyzed by some researchers in the location science ([20, 27]). It is widely considered by authors in location planning for emergency medical services (EMS) and fire stations. By applying location covering techniques within a goal programming framework, Charnes and Storbeck [8] developed a method for the sitting of multilevel EMS systems. Badri et al. [3] presented a multiple criteria modelling approach, via integer goal programming to the fire-station location problem. Alsalloum and Rand [1] considered the problem of identifying the optimal locations of a pre-specified number of emergency medical service stations using goal programming. Kanoun et al. [17] proposed a GP model to select a site for a fire and emergency service station for a case study in Tunisia.

All the above models have been formulated in a deterministic manner. In many problems however, various kinds of uncertainty and vagueness often do exist which make the models more complex. Fuzzy set theory first introduced by Zadeh [30] has been increasingly used to capture and process imprecise and uncertain information. Narasimhan [22] was the first in using fuzzy sets theory in facility location problems, albeit for decision-making about selecting gas stations not optimizing the location of a set of facilities. Recently many researchers have introduced fuzzy set theory approaches into different versions of facility location problem. Applying fuzzy set theory into GP produces Fuzzy Goal Programming (FGP) as an approach which can remove the major drawback of GP in determining precisely the goal value of each objective function. FGP first developed by Narasimhan [23] is also considered by some authors in the context of location planning. For example, Bhattacharya et al. [4] considered new facilities to be located under multiple fuzzy criteria, and developed a fuzzy goal programming approach to deal with the problems.

Bhattacharya et al. [5] formulated a fuzzy goal programming model for locating a single facility within a given convex region with the simultaneous consideration of two objectives: (i) minimize the sum of all transportation costs, and (ii) minimize the maximum distances from the facilities to the demand points.

This paper considers a special version of FGP that applies satisfaction grades instead of objective values in the context of location optimization of hierarchical public service centres.

There are some related works to our model but they all study a single level structure. Araz et al. [2] considered emergency service vehicles location problem and developed multi-objective maximal covering location model. The proposed model which uses fuzzy goal programming, allocates a fixed number of emergency service vehicles to previously defined locations so that three important service level objectives (maximization of the population covered by one vehicle, maximization of the population with backup coverage and increasing the service level by minimizing the total travel distance from locations at a distance bigger than a prespecified distance standard for all zones) can be achieved.

Tzeng and Chen [28] and Yang et al. [29] developed two similar models for sitting fire stations through a special version of fuzzy goal programming with five objectives. Aggregation was achieved through max-min operator. Genetic algorithm was utilized to solve the models.

To the best of our knowledge, this is the first work on presenting a fuzzy multi-objective programming model for locating hierarchical facilities.

3 Relevant Theory

3.1 Fuzzy Sets

In this paper, fuzzy sets theory is utilized to model uncertain travel times, cost of facilities establishment and the decision maker's preference to the partially achievement of goals.

Unlike a conventional crisp set which enforces either membership or non-membership of an object in a set, a fuzzy set allows grades of membership in the set. A fuzzy set $\tilde{A}$ is defined by a membership function $\mu_{\tilde{A}}(x)$ which assigns to each object x in the universe of discourse X, a value representing its grade of membership in this fuzzy set [30]:

$$\mu_{\tilde{A}}(x) : X \to [0,1]. \tag{1}$$

A variety of shapes such as triangular, trapezoidal, bell curves and s-curves can be used as the membership function [19]. Conventionally, the choice of the shape is subjective and allows the decision maker to express his/her preferences. A triangular fuzzy number (TFN) is denoted by triplet $\tilde{A} = (a,b,c)$ and its membership function is:

$$\mu_{\tilde{A}}(t) = \begin{cases} \dfrac{t-a}{b-a} & a \le t \le b \\ \dfrac{c-t}{c-b} & b \le t \le c \\ 0 & otherwise. \end{cases} \tag{2}$$

A typical triangular fuzzy number is shown in Figure 1.

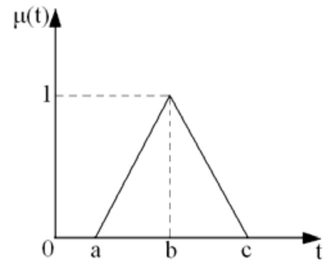

Figure 1: Triangular fuzzy number (TFN)

Since the customer doesn't have an accurate estimate of travelling times to the facilities, it is assumed that travelling times are given as linguistic terms. The linguistic approach to decision-making was chosen to formulate some uncertain aspects because it has been shown to be an effective tool for modelling qualitative information in real world decision making situations.

The linguistic terms used for travelling time variable are very soon, soon, fair, late and very late. They are defined in the universe of U=[0, $t_{\max}$], in which the variable represents the time in minutes and $t_{\max}$ is the maximum travel time on the longest path of the given area in the worst travel period.

Transformation of the fuzzy linguistic terms into triangular fuzzy numbers will be according to Table 1.

Table 1: The relationship between linguistic terms and fuzzy numbers

Fuzzy linguistic terms	Fuzzy numbers
Very Soon (VS)	(0,0, tmax/4)
Soon (S)	(0, tmax/4, tmax/2)
Fair (F)	(tmax/4, tmax/2, 3*tmax/4)
Late (L)	(tmax/2, 3*tmax/4, tmax)
Very Late (VL)	(3*tmax/4, tmax, tmax)

Similarly it is impossible to give a very accurate measure for establishment costs and the estimates usually are of approximate nature. Thus we transform the approximate measures to fuzzy numbers. The term "approximately c" can be converted into TFN with the form (c_1, c_2, c_3), as the following;

$$c_2=c \tag{3}$$

$$c_1= c.(1-(1-r).(1+s)) \tag{4}$$

$$c_3= c.(1+(1-r).(1-s)) \tag{5}$$

where $r\in[0,1]$ shows the confidence degree of DM about his/her approximation and s is the difference of right and left hand part of the membership function,

$$s= P_L-P_R \tag{6}$$

where P_L and P_R ($P_L + P_R =1$) are respectively DM's opinion about percentage of being less than c or greater than c.

For an interval approximation such as "between c_1 and c_2", it is better to convert to a trapezoidal fuzzy number. The conversion procedure is same as the above procedure.

3.2 Multi-objective Binary Programming

In a typical location model, we often deal with a series of binary decisions on locating or not locating a facility in a potential site. So, the multi-objective binary programming could be an appropriate tool for formulating our problem. The generalized representation of a multi-objective binary programming problem is as the following [11]:

$$\begin{aligned} &Min \quad Z=(f_1(X),f_2(X),...,f_P(X))^T \\ &s.t. \quad g_q(X)\le 0 \qquad q=1,2,...,Q, \\ &\qquad\; X\in\{0,1\} \end{aligned} \tag{7}$$

where $f_p(X)$, p=1,2,...,P are conflicting objective functions and $g_q(X)$, q=1,2,...,Q are constraints.

4 The Model

The model aims to answer the managerial question "How many and where should we establish hierarchical facilities to best provide our services?" The following assumptions are considered.

- The problem space is discrete so coordinates play the main role
- Euclidean distance is applied
- Customer patronizing behaviour is based on travelling time
- Facilities have a hierarchical structure with different functions
- Number of service types is equal to the number of facility levels
- Upper level facilities can offer lower level services too

The model employs the following notations:

i, I: index and set of demand points

j, J: index and set of potential facility locations

s, S: index and set of service types

k, K: index and set of facility levels

l, L: index and set of demand levels

(x,y): coordinates of a demand point or facility

$\tilde{v}_{ik}$: variable cost of establishing a facility of level k in site j

$\tilde{f}_k$: fixed costs of opening level k facility

N_s^l : number of facilities required to cover level l demand for service type s

d_{ij} : distance of demand point i to the facility in site j

$\tilde{T}_{ij}$: travel time of demand point i to the facility in site j

L_i^s : demand level of demand point i for service s, l=1, 2, 3 or 4

Decision variables are denoted by $X=\{x_{jk}\},\forall j\in J,k\in K$ which takes binary values. If a facility of level k is set up at point $j=(x,y)$, it is given by $x_{jk}=1$, otherwise, $x_{jk}=0$.

4.1 Objective Functions

As mentioned before, the objectives are minimizing average travel time for each hierarchy level, minimizing cost and maximizing adequacy of demand cover. They are detailed in the following.

Minimizing the average travel time from a demand point to a facility

According to customer' behaviour in public services, travelling time from its demand point to a facility is accounted for as his/her decision criteria in choosing the facility. The travelling time is necessarily dependent upon the distance to be travelled, the road conditions experienced during the journey and time period of the day. An individual time objective is built for each level of facilities since each level has a different accessibility time limit.

Because of different demand levels for the service provided at each level of facilities, we apply the weighted mean approach to optimize the objective. So the objective for level k of facilities' hierarchy is defined as

$$Min\ \ f_k(X)=\bar{\tilde{T}}_k=\frac{\sum_i L_i^k\otimes \underset{j,x_{jk}=1}{Min}(\tilde{T}_{ij})}{\sum_i L_i^k},\qquad k=1,2,\ldots,|K|. \tag{8}$$

As mentioned before, travelling times are assumed to be linguistic terms which could be transformed to TFN numbers. From TFN properties, the above term is also a TFN.

Minimizing the total cost

Whenever a facility is opened, the firm is faced with two main costs: building/acquisition/renting costs and the corresponding refurbishment costs with furniture and equipment, such as computers, desks, counters, air-conditioning, and security system equipments.

First category of costs for a level k facility is denoted by $\tilde{v}_{ik}$. It involves land cost and structure. It is area-dependant due to variety of land costs in different areas. Second category of costs is denoted by $\tilde{f}_k$ which is area-independent and pre-defined according to standard design plans of the firm for each hierarchy level. Accordingly, the cost objective of the model could be written as:

$$Min\ \ f_c(X)=TC=\sum_j\sum_k(\tilde{v}_{jk}+\tilde{f}_k).x_{jk}\ . \tag{9}$$

Since both categories of costs are TFN, from TFN properties, the above term is also a TFN.

Maximizing the adequacy of demand coverage

To serve appropriately a demand point with demand level l to service type s, there should be a pre-defined number of relevant facilities in its proximity. This quantity is denoted by N_s^l. Note that, opening an excessive number of such facilities doesn't necessarily improve the quality because this leads to overlapping facilities' functions, demand cannibalization and cost matters. So, it must be forced to be exactly the required number of facilities, i.e., we try to have

$$Min\ \ f_{DC}(X)=\sum_{s=1}^{S}\sum_i\Big|\,N_s^{L_i^s}-\sum_j\sum_{\substack{k=s\\x_{jk}=1}}^{K}\mu_k(d_{ij})\,\Big|\ . \tag{10}$$

The second term in (10) reflects the overall satisfaction of customers in demand point i from distance viewpoint when they request service type s,

$$u = \sum_{j} \sum_{\substack{k=s \\ x_{jk}=1}}^{K} \mu_k(d_{ij}) \tag{11}$$

where $\mu_k(d_{ij})$ is the achievement level of the distance minimization objective (not explicitly modelled here). In a minimization objective, the satisfaction of decision maker about its value is full if it is less than an optimistic value f^+, and is null if it is higher than a pessimistic value f^-. This requirement implies the fuzzy nature which can be treated by introducing the following linear membership function as the achievement level:

$$\mu(f(X)) = S_f = \begin{cases} 1 & if \quad f < f^+ \\ (f^- - f)/(f^- - f^+) & if \quad f^+ < f < f^- \\ 0 & if \quad f > f^-. \end{cases} \tag{12}$$

Such a linear membership function is illustrated in Figure 2.

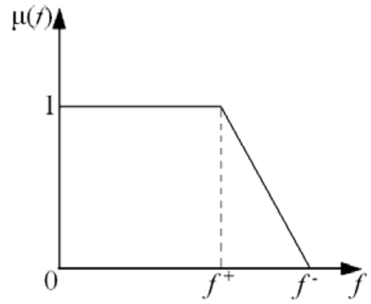

Figure 2: Achievement level of an objective

Optimistic and pessimistic values for distance of facilities to demand points are given for different service hierarchies.

Beginning index of second sigma in (11) reflects the hierarchy's nested structure. The result of (11) is between zero and total number of established facilities (M) which is unknown. It must be analyzed in relation with N_s^l. This comparison could be applied in a deterministic manner as (10). To do so in a fuzzy manner, we have

$$\mu(u) = \begin{cases} (u-a)/(N_s^l - a) & if \quad u < N_s^l \\ 1 & if \quad u = N_s^l \\ (c-u)/(c - N_s^l) & if \quad u > N_s^l \\ 0 & otherwise \end{cases} \tag{13}$$

$$a = N_s^l(1-\alpha) \tag{14}$$

$$c = N_s^l(1+\beta) \tag{15}$$

where α is a coefficient in [0,1] and its smaller values reflect DM's aversion to the demand undercover and β is a coefficient in $[0,(M - N_s^l)/N_s^l]$ and its smaller values reflect DM's aversion to the demand over-cover. Zero value for both α and β shows DM's willingness to the availability of exactly N_s^l facilities around the demand point to cover its demand for service type s.

If we extend the above process to the entire area, we have $\sum_{s=1}^{S} \sum_i N_s^{L_i^s}$ for the number of facilities required to cover all of service types for all of demand points and $\sum_{s=1}^{S} \sum_i \sum_j \sum_{k=s,x_{jk}=1}^{K} \mu_k(d_{ij})$ for the satisfaction grade of DM over the available facilities around all of demand points. So,

$$v = \sum_{s=1}^{S} \sum_i \sum_j \sum_{k=s,x_{jk}=1}^{K} \mu_k(d_{ij}), \tag{16}$$

$$N = \sum_{s=1}^{S} \sum_i N_s^{L_i^s}, \tag{17}$$

$$\mu(u)=\begin{cases}(v-a)/(N-a) & if \quad v<N\\ 1 & if \quad u=N\\ (c-v)/(c-N) & if \quad u>N\\ 0 & otherwise,\end{cases} \tag{18}$$

$$a=N(1-\alpha), \tag{19}$$

$$c=N(1+\beta) \tag{20}$$

where α is in [0,1] and β is in $[0,(S.I.M-N_s^l)/N_s^l]$.

4.2 Constraints

According to DM's view point, the first constraint may be a limitation on the total number of established facilities. Assume that the number of level k facilities must be at most M_k.

$$\sum_j x_{jk} \le M_k,\ k=1,2,\ldots,|K|. \tag{21}$$

The second constraint considers obstacles in a given area. A facility of level k should not be built up within any obstacles such as waterways, forbidden areas and reserved areas

$$x_{jk}=0, \forall j \in \theta,\ k=1,2,\ldots,|K| \tag{22}$$

where the symbol θ represents a set of obstacle coordinates.

The third set of constraints forces the model to be located a facility in some pre-defined areas. Thus,

$$x_{jk}=1, \forall j \in \phi,\ k=1,2,\ldots,|K| \tag{23}$$

where the symbol φ represents the area of before-defined locations.

There is another set of constraints through which the solution is forced to establish only one facility in each potential site, i.e.,

$$\sum_k x_{jk} \le 1, \forall j \in J. \tag{24}$$

The forth constraints set implies that the distance between any two facilities a and b of level k should be a reasonable distance, denoted by d_{ab}^k i.e. it should not be so short, as to cause overlapping of facilities functions resulting mendacious competition. Simply we use an inequality to express this constraint as

$$((x_a-x_b)^2+(y_a-y_b)^2)^{1/2} \ge d_{ab}^k \tag{25}$$

where (x_a,y_b) and (x_b,y_b) are the coordinates of the two facilities with level k located at areas a and b, respectively.

4.3 Fuzzy Multi-objective Model in a Single Unified Goal

To facilitate solving the model, its multi objectives must be integrated into a unified goal. For this purpose, all objective functions should have a unified structure and the same value range. Here we aim to convert the objective functions to their satisfaction grades. Note that, the third function, maximizing the adequacy of demand coverage is in the form of satisfaction grades.

For a fuzzy objective ($\tilde{F}$) such as average travel time from demand points to facilities and total cost which their values are fuzzy numbers, a question arises how much its goal ($\tilde{f}$) has been achieved. For example, we must compare fuzzy average travel time with its achievement level. Two approaches are investigated [12]:

- ***Possibility measure***:

The possibility measure $\mu_{\tilde{f}}(\tilde{F})$ evaluates the possibility of a fuzzy event $\tilde{F}$, occurring within the fuzzy set $\tilde{f}$. It is used to measure the satisfaction grade of a fuzzy objective value $Sg(\tilde{F})$:

$$Sg(\tilde{F})=\mu_{\tilde{f}}(\tilde{F})=\sup\min\{\mu_{\tilde{F}}(t),\mu_{\tilde{f}}(t)\} \tag{26}$$

where $\mu_{\tilde{F}}(t)$ and $\mu_{\tilde{f}}(t)$ are the membership functions of fuzzy sets $\tilde{F}$ and $\tilde{f}$ respectively.

An example of a possibility measure of fuzzy set $\tilde{F}$ with respect to fuzzy set $\tilde{f}$ is given in Figure 3.

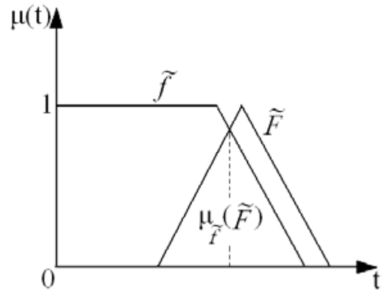

Figure 3: Satisfaction grade of fuzzy objective value using the possibility measure

- ***Area of intersection***:

Area of intersection measures the portion of $\tilde{F}$ that falls within the $\tilde{f}$ (Figure 4). Thus, the satisfaction grade of a fuzzy objective value is defined as:

$$Sg(\tilde{F}) = area(\tilde{F} \cap \tilde{f}) / area(\tilde{F}). \quad (27)$$

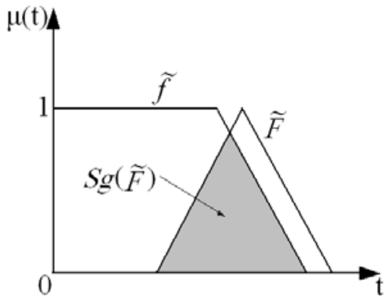

Figure 4: Satisfaction grade of fuzzy objective value using the area of intersection

The possibility measure reflects an optimistic attitude in comparison to the area of intersection because the former considers the highest point of intersection of the two fuzzy sets regardless of their overall dimensions, while the later considers the proportion of the fuzzy objective value that falls within the fuzzy achievement level.

If we optimize satisfaction grades of objective functions instead of their values, the resulted fuzzy multi-objective optimization model will be the maximization of an aggregated measure of satisfaction grades for conflicting objectives with respect to some constraints. Assuming $\mu_p(f_p(X)) = Sg(\tilde{F}_p)$, the multi-objective binary programming problem (7) is transformed to

$$\begin{aligned} &Min \quad Z = h(\mu_1(f_1(X)), \mu_2(f_2(X)), ..., \mu_p(f_P(X))) \\ &s.t. \quad g_q(X) \le 0 \qquad q = 1,2,...,Q, \\ &\qquad\; X \in \{0,1\} \end{aligned} \quad (28)$$

where h is an aggregation operator such as:

1. Simple or weighted average of the satisfaction grades:

$$Z = h(.) = \frac{1}{p}(\sum_{i=1}^{p} \mu_i(f_i(X))) \quad \text{or} \quad Z = h(.) = \frac{\sum_{i=1}^{p} w_i . \mu_i(f_i(X))}{\sum_{i=1}^{p} w_i}. \quad (29)$$

2. Geometric mean of the satisfaction grades:

$$Z = h(.) = \prod_{i=1}^{p} \mu_i(f_i(X))^{w_i}. \quad (30)$$

3. Minimum of the satisfaction grades:

$$Z = h(.) = \min_{i=1,2,...,p} \{\mu_i(f_i(X))\}. \quad (31)$$

The above aggregation operators enable DM to express his/her preferences. Average aggregation operator allows compensation for a bad value of one objective, namely a higher satisfaction grade of one objective can compensate, to a certain extent, for a lower satisfaction grade of another objective. On the contrary, minimum operator is non-compensatory, which means that the solution with a bad performance with respect to one objective will not be highly evaluated no matter how good is its performance with respect to another objectives. In the case of geometric mean, zero value for an objective leads to zero value for overall objective without regarding other objectives.

By applying the methods explained on the model objectives in Section 4.1 and with consideration of above mentioned constraints and notations, fuzzy multi-objective binary programming problem for optimization of hierarchical service centre locations can be presented as (32).

$$\begin{aligned} Max \quad & Z = h(\mu_1(f_1(X)),...,\mu_K(f_K(X)),\mu_C(f_C(X)),\mu_{DC}(f_{DC}(X))) \\ s.t. \quad & \sum_j x_{jk} = M_k, k = 1,2,...,|K| \\ & x_{jk} = 0, \forall j \in \theta, k = 1,2,...,|K| \\ & x_{jk} = 1, \forall j \in \phi, k = 1,2,...,|K| \\ & ((x_a - x_b)^2 + (y_a - y_b)^2)^{1/2} \ge d_{ab}^k, \forall a,b \in J \\ & x_{jk} \in \{0,1\}, j = 1,2,...,|J|, k = 1,2,...,|K|. \end{aligned} \tag{32}$$

It must be noted that model (32) is a special case of fuzzy goal programming without applying deviations' minimization and auxiliary variables λ. Here the satisfaction grades of the objectives are maximized.

5 Problem Solving

Although we can solve very small instances of (32) through the exact optimization methods but they are not applicable for large instances. Therefore we have to utilize approximation methods. Meta-heuristic algorithms such as Genetic Algorithms, Tabu search and Simulated Annealing have been introduced to solve these problems [28].

An important group of meta-heuristic algorithms is the local search procedures. To apply a local search to a problem instance (*S, f*) defined by a search space *S* and a profit function *f*, one first needs a neighbourhood function.

Definition Let *S* be the set of the solutions of a given instance, a neighbourhood over *S* is any function $N:S \rightarrow 2^S$. A solution *s* is a local maximum with respect to *N* if $f(s') \le f(s)$ for all $s' \in N(s)$.

We define a hybrid neighbourhood function such that it can either locate a new facility at an empty site (*Add* neighbourhood), or remove an existing facility from a site (*Drop* neighbourhood), or move a facility from a site to another site (*Interchange* neighbourhood).

The choice of neighbourhoods is probabilistic and is made by generating a random number. However, some controls are applied to avoid from blocking in the neighbour generation. For example when there is only one established facility in the current solution, the probability of selecting *Drop* neighbourhood function will be zero.

Simulated annealing and Tabu Search as the most representative local search methods are utilized to solve the problem.

5.1 Simulated Annealing

Simulated annealing (SA), introduced by Kirkpatrick et al. [18], finds its inspiration from the physical process of cooling a material to low-energy states. It repeats an iterative repairing procedure which looks for better solutions while offering the possibility of accepting worse solutions in a controlled manner. This allows SA to escape from local optima. More precisely, at each iteration, a neighbour $s' \in N(s)$ of the current solution *s* is generated randomly and a decision is then taken to decide whether *s'* will replace *s*. If *s*' is better than *s*, i.e., $\Delta = f(s') - f(s) \ge 0$ for maximization, we move from *s* to *s'*. Otherwise, we move to *s'* with the probability of $e^{\Delta/T}$. This probability depends on two factors: 1) the degree of the degradation Δ (smaller the degradation, greater the acceptance probability), and 2) a control parameter *T* called temperature (higher temperatures lead to higher acceptance probabilities and vice versa). The temperature is controlled by a cooling schedule specifying how the temperature should be progressively reduced. Typically, SA stops when a fixed number of non-improving iterations is realized or when a limit of iterations is reached. Figure 5 demonstrates pseudo-code of utilized algorithm.

The routine *InitSol*(.) gives an initial solution (s_0) in which the location of facilities is randomly generated with consideration of defined constraints. The best and current solutions of the algorithm are denoted respectively by s^* and *s*.

```
The SA Algorithm
Input: Instance G=(I, J, SerNum, travelling times, Costs,…)
Output: the best solution found (s*), Objective function value
Begin
  s0=InitSol(G)
  s*=s := s0
  MaxIteration=10.|I|.SerNum;   IterCount = (|I|. SerNum)/3
  MaxUI= MaxIteration/5;   MaxUC= IterCount/5
  Compute an initial temperature T0
  T:= T0
  While I<>MaxIteration OR Unimproved trials' number not equals to MaxUI DO
     i:=0
     While i <> IterCount OR Unchanged iterations' number not equals to MaxUC DO
        i:=i+1
        Choose randomly one of the neighborhood structures (1-Add, 2-Drop, 3-Interchange)
        Randomly choose si ∈ Nt(s)
       Δ:= f(s) − f(si)
       If Δ < 0
         s:= si
       Else
         With probability e^(−Δ/T); s:= si
       If f(s) > f(s*)
         s*:= s
     Loop
     T:= TDR.T
     I:=I+i
  Loop
End
```

Figure 5: Pseudo code of SA algorithm

After defining the initial temperature, a predefined number of iterations (*IterCount*) is executed in an inner loop in which the current solution is replaced with the generated neighbour according to SA mechanism. The algorithm is continued from outer loop with updating temperature. To efficiently guide the algorithm to escape from local optima, a stop mechanism is inserted into inner loop that terminates the loop if a predefined number of successive iterations (*MaxUC*) stay unchanged during the loop.

T is the temperature at the current iteration and is controlled by the cooling schedule function as given by $T_{IT}= TDR.T_{IT-1}$, in which IT is the current trial index and $TDR \in (0,1)$ is the temperature decreasing rate.

Stop-Condition of the algorithm is assumed to be a pre-defined number of iterations defined with respect to instance size and other difficulty parameters. Because of computational considerations, another stop condition is applied on the algorithm. If the number of failed inner loops (the loops with no improvement in the current solution) exceeds a predefined number (*MaxUI*), the algorithm terminates.

5.2 Tabu Search

Tabu Search attributed to Glover [14] is an advanced meta-heuristic algorithm with enhanced performance by using memory structures. Once a potential solution has been determined, the solution or its attributes is marked as tabu so that the solution or other solutions with same attributes are prohibited to visit for *tl* (tabu list length) next iterations. However, to mitigate the problem of losing the tabu solutions of excellent quality, aspiration criteria are introduced that is to allow selection of solutions which are better than the best-known solution. Figure 6 demonstrates the pseudo-code of utilized Tabu Search algorithm.

We define the neighbourhood function of Tabu Search as same as Simulated Annealing. The difference is in the neighbour selection such that TS selects the best possible neighbour. Due to computational challenges regarding CPU time, it is assumed that K neighbours are randomly selected from all possible neighbours and then the best non-tabu solution in the set is accepted as the new solution. The aspiration criterion is also applied.

To introduce a solution into tabu list, we need to define a solution's identification. We identify a new neighbour as <*p*, *k*> where *p* is a node index and *k* is a hierarchy level. In the case of *Add* neighbourhood function, *p* is the node selected to establish a facility and *k* is its hierarchy level. In the case of *Drop* function, *p* is the node selected to drop its established facility and *k* is its hierarchy level. In the case of *Move* function, *p* is the node selected to move its established facility to an empty node and *k* is its hierarchy level.

Once a neighbour solution is accepted, the ordered pair <*p*, *k*> is inserted into tabu list for *tl* next iterations to prevent from choosing *p* with *k* for any neighbourhood generation. Tabu list length is defined randomly from {1,2,...,20}.

```
The TS Algorithm
Input: Instance G=(I, J ,travelling times, Costs,…)
Output: the best solution found (s*), Objective function value
Begin
  s0:=InitSol(G)
  s*:=s := s0
  MaxIteration=10.|I|.SerNum;   SelNghbrCnt = (|I|. SerNum)/3
  MaxUI= MaxIteration/5
  I:=0
  While I<>MaxIteration OR Unimproved trials' number not equals to MaxUI DO
    For j =1 to SelNghbrCnt
        Choose randomly one of the neighbourhood structures (1-Add, 2-Drop, 3-Interchange)
        Randomly choose sj ϵ Nt(s)
      sj→ SelNghbrSet
    Next for
    SrtNghbrSet:=SortDes(SelNghbrSet, f(.) )
    k:=1
    SelectLable:
    s':=SrtNghbrSet (k)
    If s' IsNot Tabu OR (s' Is Tabu AND f(s')>f(s*) )
      s:= s'
    Else
      k:=k+1
      GOTO SelectLable
    Endif
    I:=I+1
    Introduce the attribute of s in the Tabu list for tl iterations
    If f(s) > f(s*)
       s*:= s
  Loop
End
```

Figure 6: Pseudo code of TS algorithm

The routine *InitSol*(.) gives an initial solution (s_0) same as the one of SA. After determining the size of candidate neighbours set, the algorithm is started from its main loop in which the best solution of the candidate neighbours set is selected as the new solution according to the TS mechanism described above.

The algorithm is continued until some conditions will be met. Stop-Condition of the algorithm is same as Simulated Annealing.

6 Computational Results

To show the algorithms' efficiency in problem solving, we have conducted some computational experiments on a set of randomly generated problems. The benchmark problems were generated as the following:

1) Each instance problem is represented by a combination of the number of nodes (n) and the number facilities' hierarchy level (s). The number of nodes was set equal to the number of a rectangular cellular board's cells which its side's length is between 5 and 10 in steps of 1. All the nodes are connected with each other. The number of facilities' level was also varied between 1 and 3. Accordingly, 63 different instances were generated.

2) Number of demand points (d) was set to [n/5] where [x] denotes the least integer number greater than or equal to x. They were located randomly in the nodes.
3) Demand level of each demand point was randomly selected from the set {1,2,3,4}.
4) Travelling time between two nodes was randomly generated from the uniform distribution [1,5].
5) Fixed cost of establishing a facility in a node was set to 5+5*k, k=1,…,s dependent on facility' level.
6) Variable cost of each facility was set to RAND[1,10]+d.(1+0.5*k), k=1,…,s.
7) Number of restricted cells to establish a facility and also number of pre-assigned cells to a facility were set to [(s-k+1)*0.01*d], k=1,…,s. They were randomly selected from the nodes.
8) The satisfaction scales for distance and travel time for each hierarchy level were defined as Table 2. Using Table 2, the optimistic and pessimistic values of the time objectives could be derived. Satisfaction of a customer about its distance to a facility could also be derived to be involved in the third objective function, maximizing the adequacy of demand coverage.

Table 2: Best range for distance and travel times

Service Hierarchy Level	Distance		Time	
	appropriate	maximum	appropriate	maximum
A	1	1.5	1.5	2.25
B	2	3	2	3
C	3	4.5	2.5	3.75

9) The minimum distance of two facilities denoted by d_{ab}^{k} was assumed to be 1, 2 and 3 respectively for hierarchy levels 1, 2 and 3.
10) Travel times between nodes were given as an n*n matrix of linguistic terms.
11) The best value for incurred total cost was set to $MCost$=40*d+RAND[0,500], while the maximum budget was set to 1.2*$MCost$.
12) The number of facilities required to cover different demand levels of a demand point was defined as Table 3.

Table 3: Number of facilities needed to cover demand levels

Facility Level \ Demand Level	1	2	3	4
1	1	2	3	4
2	1	1	2	2
3	1	1	1	2

13) The fuzzification parameters in (19) and (20) was set to $\alpha = \beta = 0.5$.

The algorithms were coded in VB and the computational experiments were carried out on a PC with 1.8 GHz Intel Dual CPU and 2 GB of RAM. The following values were set for the algorithm's parameters.

- Maximum number of iterations, $MaxIteration$=10.n.s,
- Maximum number of internal iterations for SA, $IterCount$= (n. s)/3,
- Number of candidate neighbours for TS, $SelNghbrCnt$ = (n. s)/3,
- Maximum number of successive unimproved trials, $MaxUI$= $MaxIteration$/5,
- Maximum number of successive unchanged iterations (SA), $MaxUC$= $IterCount$/5,
- Initial temperature for SA, T_0=1,
- Temperature decreasing rate for SA, TDR=0.8.

It was assumed that the number of facilities to be established is not limited, i.e., constraint (21) was not applied.

For each instance (n, s), six different configurations were constructed based on combination of comparison operators and aggregation operators. The two approaches for determining satisfaction grades of objective functions are possibility measure and area of intersection and the three aggregation operators are Arithmetic Mean, Geometric Mean and Minimum.

The proposed algorithms were run five times for each configuration and the results are reported. Table 4 gives the computational results including the average and the best objective function value, the number of objective function evaluations and the average CPU time in seconds.

Table 4: Computational results on randomly generated instances.

(x*y)	n	s	Comp. operator	Aggregation Operator	TS Average Obj.	TS Best Obj.	TS CPU Time	TS Iter. No.	SA Average Obj.	SA Best Obj.	SA CPU Time	SA Iter. No.
(5,5)	25	1	Possibility Measure	Mean	0.9986	0.9986	0.9	413	0.9983	0.9986	0.5	222
				G Mean	0.9958	0.9958	1.0	473	0.9958	0.9958	0.4	171
				Min	0.9958	0.9958	1.0	448	0.9941	0.9958	0.4	181
			Area of Intersection	Mean	0.9708	0.9708	1.0	434	0.9689	0.9708	0.5	224
				G Mean	0.9128	0.9128	1.0	468	0.9113	0.9128	0.4	189
				Min	0.9167	0.9167	0.9	412	0.9167	0.9167	0.4	164
		2	Possibility Measure	Mean	0.9997	**0.9998**	6.0	1589	0.9995	0.9996	1.9	501
				G Mean	0.9988	**0.9993**	7.1	1902	0.9979	0.9992	1.9	502
				Min	0.9988	**0.9993**	5.9	1559	0.9970	0.9981	1.9	502
			Area of Intersection	Mean	0.9770	0.9772	5.9	1547	0.9766	0.9772	1.9	502
				G Mean	0.9088	**0.9093**	6.3	1698	0.8868	0.9092	1.9	502
				Min	0.9167	0.9167	4.7	1264	0.9084	0.9167	1.9	501
		3	Possibility Measure	Mean	0.9999	**1.0000**	15.2	2892	0.9995	0.9997	4.0	752
				G Mean	0.9996	0.9999	17.4	3317	0.9966	0.9999	4.0	752
				Min	0.9991	0.9999	17.8	3399	0.9985	0.9999	4.0	751
			Area of Intersection	Mean	0.9709	**0.9770**	13.4	2528	0.9538	0.9621	4.1	752
				G Mean	0.8131	**0.8611**	18.7	3637	0.7602	0.8524	4.0	753
				Min	0.8988	0.9167	11.2	2162	0.9132	0.9167	4.0	754
(5,6)	30	1	Possibility Measure	Mean	0.9986	0.9988	2.2	756	0.9957	0.9988	0.6	218
				G Mean	0.9958	0.9965	2.2	787	0.9945	0.9965	0.5	185
				Min	0.9951	0.9965	1.9	677	0.9938	0.9965	0.6	225
			Area of Intersection	Mean	0.9322	0.9322	2.2	761	0.9315	0.9322	0.7	251
				G Mean	0.7967	0.7972	2.1	733	0.7930	0.7972	0.5	181
				Min	0.8000	0.8000	1.6	570	0.8000	0.8000	0.5	182
		2	Possibility Measure	Mean	0.9998	1.0000	14.6	3056	0.9994	1.0000	2.9	602
				G Mean	0.9996	**0.9999**	14.6	3051	0.9987	0.9994	2.8	602
				Min	0.9989	**0.9994**	9.9	2078	0.9968	0.9985	2.8	596
			Area of Intersection	Mean	0.9362	**0.9363**	10.9	2235	0.9327	0.9357	2.9	601
				G Mean	0.7560	**0.7563**	11.6	2378	0.7402	0.7552	2.9	601
				Min	0.8000	0.8000	7.2	1505	0.8000	0.8000	2.8	580
		3	Possibility Measure	Mean	0.9999	1.0000	26.5	3961	0.9997	1.0000	6.1	903
				G Mean	1.0000	**1.0000**	37.6	5664	0.9823	0.9999	6.0	902
				Min	0.9995	**1.0000**	36.5	5412	0.9982	0.9984	6.0	903
			Area of Intersection	Mean	0.9485	**0.9489**	40.9	5892	0.9368	0.9464	6.2	903
				G Mean	0.7416	**0.7562**	37.9	5457	0.6686	0.7379	6.2	902
				Min	0.8000	0.8000	21.1	3115	0.8000	0.8000	6.2	902

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS Average Obj.	TS Best Obj.	TS CPU Time	TS Iter. No.	SA Average Obj.	SA Best Obj.	SA CPU Time	SA Iter. No.
(5,7)	35	1	Possibility Measure	Mean	0.9982	0.9982	3.2	838	0.9981	0.9982	1.3	352
				G Mean	0.9951	**0.9973**	4.2	1120	0.9946	0.9946	1.2	325
				Min	0.9946	0.9946	4.7	1218	0.9937	0.9946	1.3	334
			Area of Intersection	Mean	0.9086	0.9086	3.4	879	0.9084	0.9086	1.3	351
				G Mean	0.7280	**0.7291**	4.3	1129	0.7269	0.7272	1.3	338
				Min	0.7311	0.7311	2.9	759	0.7311	0.7311	1.1	290
		2	Possibility Measure	Mean	1.0000	1.0000	21.6	3535	0.9997	1.0000	4.3	701
				G Mean	0.9998	0.9999	24.0	4026	0.9984	0.9999	4.3	702
				Min	0.9996	0.9999	25.5	4192	0.9976	0.9999	4.4	702
			Area of Intersection	Mean	0.9045	**0.9045**	19.3	3050	0.8963	0.9041	4.6	702
				G Mean	0.6480	0.6483	24.2	3874	0.6476	0.6483	4.3	702
				Min	0.7311	0.7311	13.3	2125	0.7311	0.7311	4.3	701
		3	Possibility Measure	Mean	0.9999	**1.0000**	61.7	7044	0.9997	0.9999	9.5	1055
				G Mean	0.9999	0.9999	59.4	6857	0.9981	0.9999	9.1	1053
				Min	0.9998	1.0000	73.6	8458	0.9961	1.0000	9.1	1054
			Area of Intersection	Mean	0.9069	**0.9195**	58.8	6453	0.8791	0.8912	9.4	1052
				G Mean	0.6071	**0.6233**	69.4	7754	0.5161	0.5710	9.4	1053
				Min	0.7311	0.7311	39.5	4277	0.7311	0.7311	9.4	1053
(5,8)	40	1	Possibility Measure	Mean	0.9988	0.9988	5.3	1352	0.9983	0.9988	1.6	401
				G Mean	0.9948	0.9963	5.7	1437	0.9941	0.9963	1.5	381
				Min	0.9956	0.9963	6.8	1619	0.9948	0.9963	1.5	382
			Area of Intersection	Mean	0.8796	0.8801	5.4	1351	0.8787	0.8801	1.6	401
				G Mean	0.6417	0.6417	4.6	1188	0.6408	0.6417	1.5	391
				Min	0.6441	0.6441	4.1	1027	0.6441	0.6441	1.2	309
		2	Possibility Measure	Mean	0.9997	**0.9998**	30.5	4338	0.9983	0.9996	5.7	803
				G Mean	0.9992	**0.9993**	33.6	4831	0.9962	0.9989	5.6	802
				Min	0.9990	**0.9993**	33.3	4766	0.9867	0.9989	5.6	803
			Area of Intersection	Mean	0.8569	**0.8608**	36.7	5206	0.8401	0.8537	5.7	802
				G Mean	0.5140	**0.5149**	34.6	4975	0.4851	0.5068	5.6	802
				Min	0.6441	0.6441	21.8	3103	0.6441	0.6441	5.6	801
		3	Possibility Measure	Mean	0.9999	**1.0000**	131.6	12649	0.9771	0.9994	13.3	1205
				G Mean	0.9973	**0.9998**	115.9	11312	0.9796	0.9994	12.6	1202
				Min	0.9967	**1.0000**	116.2	11578	0.9581	0.9974	12.4	1204
			Area of Intersection	Mean	0.7982	**0.8234**	85.2	8074	0.7401	0.7755	13.4	1202
				G Mean	0.3210	**0.3427**	101.9	10046	0.2490	0.2998	12.8	1205
				Min	0.6372	0.6441	93.4	9289	0.6195	0.6441	12.6	1203

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS Average Obj.	TS Best Obj.	TS CPU Time	TS Iter. No.	SA Average Obj.	SA Best Obj.	SA CPU Time	SA Iter. No.
(5,9)	45	1	Possibility Measure	Mean	0.9981	0.9981	7.6	1554	0.9984	**0.9991**	2.2	442
				G Mean	0.9944	0.9944	7.4	1617	0.9939	0.9944	2.2	433
				Min	0.9944	0.9944	6.1	1316	0.9939	0.9944	2.1	410
			Area of Intersection	Mean	0.8636	**0.8643**	6.6	1396	0.8634	0.8634	2.3	449
				G Mean	0.5927	**0.5940**	6.5	1366	0.5924	0.5924	2.3	426
				Min	0.5957	0.5957	5.8	1198	0.5957	0.5957	1.8	322
		2	Possibility Measure	Mean	0.9999	1.0000	52.9	6165	0.9997	1.0000	7.9	902
				G Mean	0.9999	0.9999	44.3	5216	0.9992	0.9999	7.7	903
				Min	0.9992	0.9999	45.9	5395	0.9993	0.9999	7.8	903
			Area of Intersection	Mean	0.8358	**0.8358**	62.2	7265	0.8293	0.8354	7.8	903
				G Mean	0.4451	0.4452	45.0	5236	0.4438	0.4452	7.9	904
				Min	0.5957	0.5957	32.5	3649	0.5957	0.5957	8.1	903
		3	Possibility Measure	Mean	1.0000	1.0000	164.9	12950	0.9995	1.0000	18.3	1353
				G Mean	0.9999	**0.9999**	158.3	12450	0.9978	0.9998	17.7	1356
				Min	0.9998	**0.9999**	179.8	14215	0.9951	0.9983	17.9	1354
			Area of Intersection	Mean	0.8407	**0.8424**	120.3	9451	0.8128	0.8321	18.7	1353
				G Mean	0.3860	**0.3869**	118.0	9366	0.3578	0.3747	18.4	1354
				Min	0.5957	0.5957	89.2	6979	0.5957	0.5957	18.2	1354
(5,10)	50	1	Possibility Measure	Mean	0.9985	0.9985	13.3	2243	0.9976	0.9985	3.0	502
				G Mean	0.9955	0.9955	12.3	2073	0.9956	**0.9977**	2.9	502
				Min	0.9959	0.9977	11.1	1887	0.9959	0.9977	2.9	501
			Area of Intersection	Mean	0.8433	0.8433	12.2	2089	0.8429	0.8433	2.9	502
				G Mean	0.5321	0.5321	11.7	2011	0.5319	0.5321	2.8	502
				Min	0.5345	0.5345	9.8	1660	0.5345	0.5345	3.0	499
		2	Possibility Measure	Mean	0.9999	**1.0000**	69.1	6800	0.9993	0.9998	10.4	1003
				G Mean	0.9996	0.9999	93.4	9268	0.9989	0.9999	10.3	1002
				Min	0.9998	**0.9999**	73.8	7290	0.9981	0.9994	10.3	1003
			Area of Intersection	Mean	0.8030	**0.8031**	81.1	8052	0.7974	0.8023	10.5	1005
				G Mean	0.3623	**0.3624**	81.0	8064	0.3591	0.3622	10.4	1004
				Min	0.5345	0.5345	46.4	4508	0.5345	0.5345	10.3	1003
		3	Possibility Measure	Mean	1.0000	1.0000	244.6	16079	0.9988	1.0000	34.8	1506
				G Mean	0.9999	**1.0000**	246.8	16664	0.9862	0.9989	34.2	1503
				Min	0.9999	**1.0000**	197.4	13336	0.9925	0.9987	33.9	1506
			Area of Intersection	Mean	0.7984	**0.8017**	263.0	17610	0.7479	0.7877	35.9	1504
				G Mean	0.2871	**0.2892**	317.5	21429	0.2365	0.2599	35.3	1506
				Min	0.5345	0.5345	144.9	8802	0.5345	0.5345	34.3	1502

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS Average Obj.	TS Best Obj.	TS CPU Time	TS Iter. No.	SA Average Obj.	SA Best Obj.	SA CPU Time	SA Iter. No.
(6,6)	36	1	Possibility Measure	Mean	0.9986	**0.9990**	4.2	1008	0.9978	0.9980	1.6	359
				G Mean	0.9953	0.9971	4.2	1034	0.9953	0.9971	1.3	321
				Min	0.9959	0.9971	4.7	1147	0.9937	0.9971	1.4	326
			Area of Intersection	Mean	0.8988	**0.8990**	4.1	984	0.8980	0.8980	1.6	361
				G Mean	0.6975	0.6979	4.5	1125	0.6963	0.6979	1.5	347
				Min	0.7000	0.7000	3.3	811	0.7000	0.7000	1.2	285
		2	Possibility Measure	Mean	0.9999	1.0000	21.2	3020	0.9998	1.0000	5.2	723
				G Mean	0.9999	0.9999	24.7	3538	0.9994	0.9999	5.1	721
				Min	0.9999	0.9999	36.1	5066	0.9995	0.9999	5.1	722
			Area of Intersection	Mean	0.8892	**0.8893**	24.6	3440	0.8891	0.8891	5.2	722
				G Mean	0.5999	0.6000	20.9	2944	0.5994	0.6000	5.2	722
				Min	0.7000	0.7000	15.6	2203	0.7000	0.7000	5.2	722
		3	Possibility Measure	Mean	1.0000	**1.0000**	69.1	7127	0.9999	1.0000	11.3	1083
				G Mean	0.9999	**0.9999**	77.7	7916	0.9986	0.9998	10.8	1084
				Min	0.9999	**1.0000**	52.1	5364	0.9982	0.9997	10.7	1084
			Area of Intersection	Mean	0.9036	**0.9037**	86.3	7253	0.8858	0.8942	11.1	1083
				G Mean	0.5766	**0.5769**	72.8	6069	0.5608	0.5765	11.1	1082
				Min	0.7000	0.7000	41.8	4128	0.7000	0.7000	11.2	1083
(6,7)	42	1	Possibility Measure	Mean	0.9368	0.9393	12.8	1488	0.9299	**0.9414**	2.0	421
				G Mean	0.8297	0.8342	11.7	1493	0.8333	**0.8469**	1.9	421
				Min	0.8473	0.8748	14.4	1610	0.8659	**0.8969**	1.9	421
			Area of Intersection	Mean	0.7849	**0.7939**	17.9	1795	0.7295	0.7938	1.9	421
				G Mean	0.4601	**0.4763**	21.2	1849	0.4475	0.4632	1.8	420
				Min	0.6190	0.6190	16.5	1306	0.6190	0.6190	1.9	416
		2	Possibility Measure	Mean	0.9985	**0.9997**	88.0	6305	0.9443	0.9833	6.9	841
				G Mean	0.9971	**0.9990**	79.0	6393	0.9449	0.9792	6.6	841
				Min	0.9811	**0.9986**	104.2	6126	0.8622	0.9728	6.7	842
			Area of Intersection	Mean	0.8292	**0.8411**	57.8	6533	0.6995	0.7755	7.3	843
				G Mean	0.4343	**0.4633**	51.3	6819	0.3324	0.4111	6.7	844
				Min	0.6190	0.6190	28.8	3830	0.6065	0.6190	6.5	844
		3	Possibility Measure	Mean	0.9976	**0.9995**	116.0	10311	0.9717	0.9907	16.5	1264
				G Mean	0.9784	**0.9928**	134.4	11970	0.9044	0.9918	15.8	1262
				Min	0.9862	**0.9930**	110.6	9856	0.9039	0.9540	16.4	1263
			Area of Intersection	Mean	0.8020	**0.8257**	141.9	12119	0.6947	0.7345	18.0	1265
				G Mean	0.3058	**0.3331**	148.9	13093	0.1748	0.2070	17.5	1265
				Min	0.6190	0.6190	86.2	7711	0.6041	0.6190	16.4	1263

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS Average Obj.	TS Best Obj.	TS CPU Time	TS Iter. No.	SA Average Obj.	SA Best Obj.	SA CPU Time	SA Iter. No.
(6,8)	48	1	Possibility Measure	Mean	0.9987	0.9991	8.8	1483	0.9987	0.9991	2.9	481
				G Mean	0.9951	0.9973	9.0	1509	0.9957	0.9973	3.1	482
				Min	0.9968	0.9973	8.2	1425	0.9957	0.9973	3.0	482
			Area of Intersection	Mean	0.8499	0.8503	8.6	1453	0.8496	0.8503	3.0	481
				G Mean	0.5518	0.5521	10.5	1757	0.5512	0.5521	2.9	481
				Min	0.5536	0.5536	8.5	1436	0.5536	0.5536	2.8	450
		2	Possibility Measure	Mean	0.9999	0.9999	51.8	4845	0.9998	0.9999	10.6	962
				G Mean	0.9997	0.9999	63.0	5896	0.9994	0.9999	10.6	963
				Min	0.9994	0.9996	55.8	5244	0.9995	**0.9999**	10.6	962
			Area of Intersection	Mean	0.8134	**0.8134**	83.3	7697	0.8129	0.8132	10.5	964
				G Mean	0.3874	**0.3875**	57.2	5376	0.3872	0.3873	10.4	964
				Min	0.5536	0.5536	43.3	3969	0.5536	0.5536	10.6	963
		3	Possibility Measure	Mean	1.0000	1.0000	208.0	11843	0.9997	1.0000	23.7	1443
				G Mean	1.0000	**1.0000**	248.9	13888	0.9994	0.9998	23.7	1446
				Min	1.0000	**1.0000**	235.8	14927	0.9996	0.9999	23.6	1445
			Area of Intersection	Mean	0.8151	**0.8152**	215.3	13518	0.7946	0.8125	34.0	1445
				G Mean	0.3187	**0.3187**	238.7	14852	0.2954	0.3183	23.7	1447
				Min	0.5536	0.5536	127.5	7523	0.5536	0.5536	23.9	1445
(6,9)	54	1	Possibility Measure	Mean	0.9984	0.9985	14.6	2037	0.9985	0.9985	3.9	542
				G Mean	0.9956	0.9956	17.1	2399	0.9960	**0.9978**	4.0	541
				Min	0.9960	0.9978	17.1	2359	0.9953	0.9978	3.9	542
			Area of Intersection	Mean	0.8320	**0.8326**	17.4	2342	0.8319	0.8319	3.8	541
				G Mean	0.4980	0.4989	17.0	2327	0.4978	0.4989	3.8	541
				Min	0.5000	0.5000	13.5	1819	0.5000	0.5000	3.8	519
		2	Possibility Measure	Mean	0.9999	1.0000	99.1	7942	0.9998	1.0000	13.8	1083
				G Mean	0.9996	0.9999	94.5	7691	0.9997	0.9999	13.7	1084
				Min	0.9994	0.9999	109.4	8754	0.9996	0.9999	13.8	1084
			Area of Intersection	Mean	0.7843	**0.7844**	104.4	8397	0.7829	0.7843	13.6	1084
				G Mean	0.3187	**0.3188**	104.7	8514	0.3179	0.3187	13.6	1084
				Min	0.5000	0.5000	66.1	5238	0.5000	0.5000	13.8	1083
		3	Possibility Measure	Mean	1.0000	**1.0000**	278.4	15339	0.9996	0.9997	31.1	1623
				G Mean	0.9999	**1.0000**	314.4	17409	0.9989	0.9998	31.9	1625
				Min	1.0000	**1.0000**	302.2	16599	0.9982	0.9995	30.5	1625
			Area of Intersection	Mean	0.7789	**0.7790**	326.6	18213	0.7535	0.7641	31.1	1622
				G Mean	0.2414	**0.2415**	261.1	14617	0.2266	0.2403	31.0	1627
				Min	0.5000	0.5000	185.3	10053	0.5000	0.5000	31.2	1626

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS				SA			
					Average Obj.	Best Obj.	CPU Time	Iter. No.	Average Obj.	Best Obj.	CPU Time	Iter. No.
(6,10)	60	1	Possibility Measure	Mean	0.9985	**0.9988**	23.9	3010	0.9983	0.9987	5.1	601
				G Mean	0.9962	**0.9964**	23.1	2932	0.9944	0.9944	4.9	602
				Min	0.9962	0.9964	22.7	2854	0.9951	0.9964	5.0	601
			Area of Intersection	Mean	0.8172	**0.8172**	20.7	2639	0.8169	0.8172	5.1	602
				G Mean	0.4539	0.4539	24.3	3143	0.4535	0.4539	4.8	602
				Min	0.4556	0.4556	17.9	2251	0.4556	0.4556	4.8	571
		2	Possibility Measure	Mean	1.0000	1.0000	195.4	14068	0.9999	1.0000	18.2	1204
				G Mean	0.9999	0.9999	164.6	9433	0.9990	0.9999	17.8	1203
				Min	0.9997	0.9999	120.4	8248	0.9993	0.9999	17.7	1203
			Area of Intersection	Mean	0.7600	**0.7600**	131.6	9597	0.7582	0.7598	18.0	1203
				G Mean	0.2663	**0.2663**	162.0	11786	0.2658	0.2662	17.6	1203
				Min	0.4556	0.4556	93.4	6522	0.4556	0.4556	17.8	1205
		3	Possibility Measure	Mean	1.0000	1.0000	464.1	22018	0.9993	1.0000	40.7	1806
				G Mean	1.0000	**1.0000**	419.4	19755	0.9995	0.9997	39.5	1805
				Min	1.0000	**1.0000**	569.5	26736	0.9943	0.9996	39.7	1807
			Area of Intersection	Mean	0.7477	**0.7479**	501.9	22346	0.7229	0.7396	41.1	1807
				G Mean	0.1863	**0.1864**	552.0	24045	0.1736	0.1838	40.9	1804
				Min	0.4556	0.4556	277.5	12911	0.4556	0.4556	41.0	1801
(7,7)	49	1	Possibility Measure	Mean	0.9986	**0.9992**	14.5	2303	0.9984	0.9984	3.2	491
				G Mean	0.9962	**0.9976**	13.3	2089	0.9953	0.9953	3.4	491
				Min	0.9953	0.9953	11.0	1789	0.9941	0.9953	3.2	493
			Area of Intersection	Mean	0.8498	**0.8504**	12.6	1998	0.8495	0.8496	3.2	492
				G Mean	0.5513	**0.5523**	13.7	2222	0.5508	0.5510	3.2	492
				Min	0.5536	0.5536	9.7	1576	0.5536	0.5536	3.1	471
		2	Possibility Measure	Mean	1.0000	1.0000	76.5	7574	0.9999	1.0000	10.6	983
				G Mean	0.9995	0.9999	52.2	5132	0.9997	0.9999	10.3	984
				Min	0.9999	0.9999	81.5	7959	0.9998	0.9999	10.4	983
			Area of Intersection	Mean	0.8134	0.8134	85.3	8169	0.8133	0.8134	10.5	983
				G Mean	0.3875	0.3875	86.1	8325	0.3871	0.3875	10.6	984
				Min	0.5536	0.5536	47.9	4265	0.5536	0.5536	10.9	984
		3	Possibility Measure	Mean	1.0000	1.0000	261.4	16632	0.9999	1.0000	24.6	1475
				G Mean	1.0000	**1.0000**	243.3	15520	0.9995	0.9999	23.9	1473
				Min	0.9999	**1.0000**	232.7	14831	0.9995	0.9999	24.2	1473
			Area of Intersection	Mean	0.8152	**0.8152**	164.9	10518	0.8091	0.8149	24.5	1475
				G Mean	0.3186	**0.3187**	256.3	16295	0.3072	0.3186	24.8	1473
				Min	0.5536	0.5536	131.6	8129	0.5536	0.5536	25.1	1474

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS Average Obj.	TS Best Obj.	TS CPU Time	TS Iter. No.	SA Average Obj.	SA Best Obj.	SA CPU Time	SA Iter. No.
(7,8)	56	1	Possibility Measure	Mean	0.9985	0.9986	21.5	2945	0.9985	0.9986	4.2	562
				G Mean	0.9958	0.9958	20.1	2762	0.9958	0.9958	4.2	562
				Min	0.9958	0.9958	16.5	2260	0.9948	0.9958	4.4	562
			Area of Intersection	Mean	0.8267	0.8267	18.5	2586	0.8266	0.8267	4.3	562
				G Mean	0.4822	0.4822	18.5	2604	0.4822	0.4822	4.2	561
				Min	0.4843	0.4843	15.7	2127	0.4843	0.4843	4.1	541
		2	Possibility Measure	Mean	1.0000	**1.0000**	111.5	7814	0.9997	0.9999	15.1	1122
				G Mean	0.9999	**0.9999**	129.2	10059	0.9988	0.9995	15.1	1124
				Min	0.9998	0.9999	92.8	7238	0.9997	0.9999	15.2	1124
			Area of Intersection	Mean	0.7758	**0.7758**	129.4	10082	0.7669	0.7751	15.2	1122
				G Mean	0.2997	**0.2998**	122.3	9505	0.2920	0.2997	15.1	1123
				Min	0.4843	0.4843	75.9	5835	0.4843	0.4843	15.0	1122
		3	Possibility Measure	Mean	0.9999	**1.0000**	553.3	28267	0.9755	0.9894	36.4	1688
				G Mean	0.9988	**0.9997**	325.6	16661	0.9501	0.9954	34.6	1686
				Min	0.9947	**0.9997**	558.3	28639	0.9035	0.9612	34.6	1684
			Area of Intersection	Mean	0.7344	**0.7484**	388.3	19578	0.5969	0.6440	38.3	1688
				G Mean	0.1759	**0.1862**	445.4	22470	0.1195	0.1329	35.8	1685
				Min	0.4843	0.4843	262.1	11729	0.4843	0.4843	35.2	1687
(7,9)	63	1	Possibility Measure	Mean	0.9988	**0.9993**	23.0	2839	0.9969	0.9985	5.5	631
				G Mean	0.9965	**0.9978**	28.1	3490	0.9894	0.9956	5.3	632
				Min	0.9965	0.9978	30.1	3717	0.9927	0.9978	5.3	632
			Area of Intersection	Mean	0.8114	**0.8127**	34.8	4376	0.7654	0.7988	5.6	632
				G Mean	0.4393	0.4414	31.9	4017	0.4279	0.4412	5.2	633
				Min	0.4424	0.4424	20.4	2488	0.4424	0.4424	5.3	631
		2	Possibility Measure	Mean	0.9514	**0.9538**	289.3	19023	0.8868	0.9293	22.4	1267
				G Mean	0.7975	**0.8234**	248.5	16290	0.7287	0.7666	26.4	1264
				Min	0.8191	**0.8323**	176.2	11196	0.7152	0.7664	32.1	1263
			Area of Intersection	Mean	0.6465	**0.6490**	233.1	15013	0.5034	0.5046	22.8	1262
				G Mean	0.1492	**0.1538**	233.3	15489	0.1210	0.1311	19.8	1266
				Min	0.4424	0.4424	117.6	7611	0.4424	0.4424	19.8	1265
		3	Possibility Measure	Mean	0.9994	**0.9999**	876.0	26841	0.9145	0.9382	48.4	1896
				G Mean	0.9968	**0.9993**	529.8	23631	0.7593	0.8187	45.4	1897
				Min	0.9738	**0.9981**	618.1	27455	0.7191	0.8614	44.9	1897
			Area of Intersection	Mean	0.6989	**0.7100**	640.3	28103	0.5384	0.5427	51.6	1901
				G Mean	0.1359	**0.1424**	523.1	21772	0.0538	0.0625	46.8	1895
				Min	0.4424	0.4424	364.1	16131	0.4217	0.4424	45.4	1896

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS Average Obj.	TS Best Obj.	TS CPU Time	TS Iter. No.	SA Average Obj.	SA Best Obj.	SA CPU Time	SA Iter. No.
(7,10)	70	1	Possibility Measure	Mean	0.9987	0.9989	40.1	4096	0.9986	0.9989	7.4	701
				G Mean	0.9965	0.9965	45.1	4636	0.9957	0.9965	7.1	702
				Min	0.9965	0.9965	35.6	3652	0.9959	0.9965	7.2	701
			Area of Intersection	Mean	0.7977	0.7977	37.1	3901	0.7817	0.7977	7.4	702
				G Mean	0.3951	0.3951	36.0	3785	0.3947	0.3951	7.1	701
				Min	0.3964	0.3964	31.9	3187	0.3964	0.3964	7.4	701
		2	Possibility Measure	Mean	0.9999	**1.0000**	332.4	18146	0.9752	0.9892	28.2	1406
				G Mean	0.9996	**0.9999**	267.2	14849	0.9339	0.9693	26.9	1404
				Min	0.9854	**0.9994**	222.3	12202	0.8830	0.9377	27.0	1405
			Area of Intersection	Mean	0.7186	**0.7263**	329.8	18576	0.5796	0.6583	29.7	1413
				G Mean	0.1899	**0.1965**	348.7	19568	0.1429	0.1535	27.3	1405
				Min	0.3964	0.3964	168.8	9236	0.3964	0.3964	26.7	1407
		3	Possibility Measure	Mean	1.0000	**1.0000**	952.2	29611	0.9925	0.9995	65.2	2112
				G Mean	1.0000	**1.0000**	914.9	34350	0.9914	0.9998	60.9	2106
				Min	0.9999	**1.0000**	808.4	30217	0.9411	0.9829	60.1	2107
			Area of Intersection	Mean	0.7024	**0.7054**	885.4	32777	0.5353	0.6008	67.0	2116
				G Mean	0.1234	**0.1251**	1058.1	39406	0.0896	0.0977	62.6	2107
				Min	0.3964	0.3964	482.8	17795	0.3964	0.3964	59.9	2105
(8,8)	64	1	Possibility Measure	Mean	0.9467	**0.9500**	41.8	5244	0.9325	0.9402	10.1	642
				G Mean	0.8461	**0.8597**	37.6	4696	0.8205	0.8346	10.6	642
				Min	0.8805	**0.9014**	39.3	4766	0.8179	0.8597	10.1	643
			Area of Intersection	Mean	0.7254	**0.7278**	32.7	4246	0.6530	0.6957	9.6	642
				G Mean	0.3294	**0.3336**	35.5	4599	0.2367	0.3005	10.4	642
				Min	0.4300	0.4300	23.6	2932	0.4245	0.4300	9.7	642
		2	Possibility Measure	Mean	0.9995	**0.9998**	220.8	15120	0.9789	0.9865	23.3	1284
				G Mean	0.9994	**0.9999**	194.4	13395	0.9454	0.9830	23.5	1282
				Min	0.9979	**0.9999**	186.8	12745	0.9396	0.9682	23.6	1284
			Area of Intersection	Mean	0.7297	**0.7336**	193.6	13336	0.5399	0.6687	23.9	1281
				G Mean	0.2236	**0.2262**	227.9	15777	0.1654	0.1819	21.5	1284
				Min	0.4300	0.4300	113.0	7734	0.4300	0.4300	21.3	1284
		3	Possibility Measure	Mean	0.9986	**0.9994**	798.3	34584	0.9372	0.9421	50.5	1927
				G Mean	0.9958	**0.9984**	586.5	26026	0.7894	0.8518	77.4	1928
				Min	0.9624	**0.9899**	649.9	28336	0.6687	0.7939	50.9	1928
			Area of Intersection	Mean	0.6693	**0.6847**	641.6	28256	0.5254	0.5409	52.1	1925
				G Mean	0.1088	**0.1134**	673.6	29810	0.0486	0.0558	50.1	1927
				Min	0.4300	0.4300	384.9	17101	0.3960	0.4300	48.2	1926

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS				SA			
					Average Obj.	Best Obj.	CPU Time	Iter. No.	Average Obj.	Best Obj.	CPU Time	Iter. No.
(8,9)	72	1	Possibility Measure	Mean	0.9744	**0.9777**	48.9	4669	0.9658	0.9727	7.8	723
				G Mean	0.9251	**0.9356**	51.7	4869	0.9102	0.9269	7.7	722
				Min	0.9298	**0.9345**	59.8	5538	0.8837	0.9044	8.1	723
			Area of Intersection	Mean	0.7408	**0.7429**	61.9	5993	0.7110	0.7343	7.6	723
				G Mean	0.3254	**0.3280**	57.8	5532	0.3118	0.3180	7.5	722
				Min	0.3864	**0.3864**	37.7	3524	0.3864	0.3864	7.9	722
		2	Possibility Measure	Mean	0.9965	**1.0000**	285.6	14779	0.9405	0.9784	31.3	1443
				G Mean	0.9950	**0.9977**	328.9	17026	0.9284	0.9689	29.3	1444
				Min	0.9758	**0.9915**	478.9	24670	0.8878	0.9263	30.0	1445
			Area of Intersection	Mean	0.6917	**0.7005**	333.3	17527	0.5176	0.6372	32.8	1441
				G Mean	0.1753	**0.1799**	531.5	25956	0.1393	0.1503	30.1	1445
				Min	0.3864	**0.3864**	186.0	9662	0.3864	0.3864	29.2	1443
		3	Possibility Measure	Mean	1.0000	**1.0000**	1203.8	41538	0.9786	0.9814	71.4	2168
				G Mean	0.9992	**0.9999**	1047.4	36267	0.9242	0.9667	67.7	2166
				Min	0.9940	**0.9999**	971.7	33520	0.8698	0.9217	68.4	2170
			Area of Intersection	Mean	0.6715	**0.6817**	1032.2	35832	0.4972	0.4994	75.8	2168
				G Mean	0.0959	**0.1073**	1064.0	37023	0.0619	0.0723	71.2	2171
				Min	0.3864	**0.3864**	564.6	19540	0.3864	0.3864	70.2	2167
(8,10)	80	1	Possibility Measure	Mean	0.9989	**0.9989**	48.9	4327	0.9953	0.9989	10.3	802
				G Mean	0.9969	**0.9983**	53.7	4746	0.9960	0.9966	9.6	803
				Min	0.9969	**0.9983**	51.6	4520	0.9955	0.9966	9.7	802
			Area of Intersection	Mean	0.7824	**0.7824**	47.2	4201	0.7410	0.7824	10.5	802
				G Mean	0.3495	**0.3495**	52.1	4683	0.3425	0.3495	10.5	803
				Min	0.3507	**0.3507**	49.3	4157	0.3507	0.3507	9.7	802
		2	Possibility Measure	Mean	1.0000	**1.0000**	420.4	19835	0.9131	0.9841	41.2	1603
				G Mean	0.9999	**0.9999**	469.1	21864	0.9639	0.9988	37.0	1606
				Min	0.9999	**0.9999**	689.1	21208	0.8810	0.9352	37.5	1607
			Area of Intersection	Mean	0.7004	**0.7008**	462.7	21725	0.4612	0.4774	42.6	1604
				G Mean	0.1581	**0.1594**	560.7	26348	0.1196	0.1308	37.0	1607
				Min	0.3507	**0.3507**	313.1	12343	0.3507	0.3507	38.8	1606
		3	Possibility Measure	Mean	0.9988	**0.9998**	1247.0	38178	0.9741	0.9844	91.8	2411
				G Mean	0.9940	**0.9998**	1788.6	54861	0.7782	0.9417	89.5	2411
				Min	0.9940	**0.9982**	1503.5	45981	0.8174	0.9194	95.3	2404
			Area of Intersection	Mean	0.6341	**0.6397**	1724.2	53052	0.4719	0.4727	99.5	2411
				G Mean	0.0685	**0.0724**	1548.8	47612	0.0339	0.0533	93.4	2410
				Min	0.3507	**0.3507**	801.2	24331	0.2530	0.3507	100.3	2406

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS Average Obj.	TS Best Obj.	TS CPU Time	TS Iter. No.	SA Average Obj.	SA Best Obj.	SA CPU Time	SA Iter. No.
(9,9)	81	1	Possibility Measure	Mean	0.9991	**0.9991**	62.7	4801	0.9986	0.9986	11.1	813
				G Mean	0.9972	**0.9974**	97.7	7499	0.9959	0.9959	10.7	813
				Min	0.9969	**0.9974**	67.0	5193	0.9962	0.9974	10.8	813
			Area of Intersection	Mean	0.7826	0.7826	70.6	5605	0.7823	0.7826	10.7	814
				G Mean	0.3497	**0.3498**	61.7	4911	0.3473	0.3493	11.1	813
				Min	0.3507	0.3507	56.2	4233	0.3507	0.3507	11.1	812
		2	Possibility Measure	Mean	1.0000	**1.0000**	452.6	18336	0.9977	0.9999	42.6	1627
				G Mean	1.0000	**1.0000**	559.5	23065	0.9992	0.9996	40.8	1625
				Min	0.9998	**0.9998**	460.2	18988	0.9950	0.9992	40.6	1624
			Area of Intersection	Mean	0.7014	**0.7015**	490.2	20225	0.6702	0.6985	44.3	1632
				G Mean	0.1597	**0.1597**	614.3	23964	0.1529	0.1594	42.1	1625
				Min	0.3507	0.3507	312.1	12381	0.3507	0.3507	42.5	1625
		3	Possibility Measure	Mean	1.0000	**1.0000**	2006.1	55561	0.9967	0.9999	101.6	2440
				G Mean	1.0000	**1.0000**	1877.0	52048	0.9846	0.9990	95.4	2436
				Min	0.9999	**1.0000**	1717.7	47597	0.9487	0.9998	98.2	2437
			Area of Intersection	Mean	0.6701	**0.6718**	2228.5	62182	0.5397	0.6271	105.9	2455
				G Mean	0.0867	**0.0878**	1826.9	50990	0.0615	0.0694	96.7	2443
				Min	0.3507	0.3507	896.5	24476	0.3507	0.3507	98.1	2436
(9,10)	90	1	Possibility Measure	Mean	0.9991	0.9992	114.9	8014	0.9084	0.9992	15.0	903
				G Mean	0.9976	0.9976	120.6	8328	0.9781	0.9976	13.3	903
				Min	0.9957	**0.9976**	138.0	9438	0.9048	0.9429	13.9	902
			Area of Intersection	Mean	0.7688	**0.7705**	176.4	11835	0.6198	0.7522	15.1	901
				G Mean	0.3108	**0.3135**	178.1	8610	0.2761	0.2910	13.7	905
				Min	0.3143	0.3143	83.3	5577	0.3143	0.3143	13.9	902
		2	Possibility Measure	Mean	1.0000	1.0000	566.5	18744	0.9992	1.0000	54.0	1806
				G Mean	0.9999	**1.0000**	568.8	21285	0.9991	0.9998	51.5	1807
				Min	0.9999	1.0000	681.3	25137	0.9988	1.0000	50.9	1804
			Area of Intersection	Mean	0.6809	**0.6810**	792.3	28277	0.6720	0.6793	54.9	1809
				G Mean	0.1288	0.1288	605.5	21260	0.1282	0.1288	51.7	1806
				Min	0.3143	0.3143	417.5	15081	0.3143	0.3143	55.2	1805
		3	Possibility Measure	Mean	1.0000	1.0000	1931.5	47362	0.9977	1.0000	122.6	2707
				G Mean	1.0000	**1.0000**	1954.6	48324	0.9997	0.9999	121.4	2706
				Min	1.0000	**1.0000**	2253.6	55564	0.9925	0.9998	118.4	2713
			Area of Intersection	Mean	0.6447	**0.6448**	2943.6	73412	0.5918	0.6233	133.2	2706
				G Mean	0.0644	**0.0644**	2696.5	67042	0.0461	0.0634	131.5	2708
				Min	0.3143	0.3143	1345.7	30583	0.3143	0.3143	148.3	2711

Table 4: (*continued*)

(x*y)	n	s	Comp. operator	Aggregation Operator	TS				SA			
					Average Obj.	Best Obj.	CPU Time	Iter. No.	Average Obj.	Best Obj.	CPU Time	Iter. No.
(10,10)	100	1	Possibility Measure	Mean	0.9407	**0.9415**	194.5	11036	0.9260	0.9395	18.2	1003
				G Mean	0.8257	**0.8307**	218.0	12422	0.8157	0.8246	18.2	1003
				Min	0.8660	**0.8776**	197.4	10946	0.7933	0.8419	19.3	1004
			Area of Intersection	Mean	0.6734	**0.6748**	175.3	10345	0.5270	0.6668	21.6	1003
				G Mean	0.2118	**0.2123**	190.6	10967	0.1617	0.2045	19.1	1003
				Min	0.2847	0.2847	127.9	6991	0.2847	0.2847	18.9	1001
		2	Possibility Measure	Mean	0.9998	**1.0000**	928.5	28129	0.9674	0.9901	77.2	2010
				G Mean	0.9996	**1.0000**	900.6	27375	0.9672	0.9970	70.1	2006
				Min	0.9979	**0.9996**	1380.5	41872	0.9053	0.9401	71.2	2008
			Area of Intersection	Mean	0.6563	**0.6580**	1507.4	46250	0.4285	0.4406	81.3	2007
				G Mean	0.1016	**0.1033**	1491.0	45818	0.0883	0.0957	72.5	2010
				Min	0.2847	0.2847	624.2	18707	0.2847	0.2847	71.0	2005
		3	Possibility Measure	Mean	0.9922	**0.9950**	4504.2	88127	0.8170	0.8700	188.7	3034
				G Mean	0.9644	**0.9793**	4610.2	90163	0.5418	0.8121	183.2	3010
				Min	0.9086	**0.9312**	3445.5	60598	0.5985	0.7653	187.9	3015
			Area of Intersection	Mean	0.5743	**0.5791**	4119.3	81864	0.4224	0.4225	184.9	3009
				G Mean	0.0362	**0.0378**	4961.7	98801	0.0061	0.0146	194.5	3011
				Min	0.2847	0.2847	1952.8	38228	0.1615	0.2847	197.5	3010

From Table 3 we can see that in terms of solution quality, Tabu Search performs better than Simulated Annealing in 54% of instances and Simulated Annealing is better than Tabu Search in 2% of instances. This is while they are the same in other instances. Note that the best objective function value is boldfaced for the better algorithm. The two last columns for each algorithm show its efficiency in terms of iteration numbers and computational time in seconds. From this viewpoint, SA is better than TS in all instances. Therefore, it can be concluded that TS is a high quality while slow algorithm and SA is a fast while low quality algorithm in our considered problem.

Although the values in each row cannot be compared with another row due to the difference in the nature of the aggregation operators, the values resulted from possibility measure are usually larger than those from area of intersection. This is because of pessimistic attitude of DM when choosing the area of intersection approach.

To decide an aggregation operator, it is required to analyse its outcomes. To do so, the single objective values obtained through an operator is aggregated using the other ones. The results show that the use of geometric mean aggregation operator optimizes also the two other aggregation operators.

7 Conclusion and Future Research

Determination of optimal number and location of hierarchical facilities has been considered in this paper. Three types of objectives have been defined: minimizing the average travel times, minimizing the total costs and maximizing the adequacy of demand coverage. This choice is because that travel time has an important role in emergency service organizations such as hospitals and police offices and cost minimization is a common objective in location decisions. The adequacy of demand coverage is applied to formulate capacity constraints using a different language through which we limit over-cover and under-cover situations of demand points. A rich set of constraints have also been defined: two subsets consider obstacles in a given area, and the other concerns the distance between any two adjacent facilities in a level.

Instead of accurate estimation of amount of customers' demand, the demand points have been assumed to be of four levels. This is the case for emergency service systems.

Due to inevitable uncertainties in the problem (e.g. travel times and costs) fuzzy sets theory has been utilized to build a model to fit fairly real-world circumstances.

For each objective function, a satisfaction grade has been defined by which DM could express his/her preference to the partially achievement of objectives. Satisfaction grade is obtainable through two approaches as possibility measure and area of intersection. Based on the satisfaction grades of objectives, we have converted fuzzy multi-objective optimization model into a single unified goal through three different aggregation operators.

Two efficient local search meta-heuristics (Simulated Annealing and Tabu Search) have been applied to solve the problem. Efficiency of the algorithms has been demonstrated through a rich set of experiments. Various configurations of the model have been considered in the experiments.

The model provides some capabilities. It studies hierarchical structure of facilities that is rarely considered in the literature. The optimal number of facilities is generated by the model itself. The model utilizes satisfaction grades in the context of fuzzy multi objective programming. It applies satisfaction grades of objective function values instead of their mere values.

Future work on the related problems mainly considers how to model and deal with consideration of waiting times at facilities. The facilities in each level may be of different sizes or different design attributes.

References

[1] Alsalloum, O.I., and G.K. Rand, Extensions to emergency vehicle location models, *Computers and Operations Research,* vol.33, no.9, pp.2725–2743, 2006.

[2] Araz, C., H. Selim, and I. Ozkarahan, A fuzzy multi-objective covering-based vehicle location model for emergency services, *Computers and Operations Research,* vol.34, pp.705–726, 2007.

[3] Badri, M.A., A.M. Mortagy, and C.A. Alsayed, A multi-objective model for locating fire stations, *European Journal of Operational Research,* vol.110, pp.243–260, 1998.

[4] Bhattacharya, U., J.R. Rao, and R.N. Tiwari, Fuzzy multi-criteria facility location problem, *Fuzzy Sets and Systems*, vol.51, pp.277–287, 1992.

[5] Bhattacharya, U., J.R. Rao, and R.N. Tiwari, Bi-criteria multi facility location problem in fuzzy environment, *Fuzzy Sets and Systems*, vol.56, no.2, pp.145–153, 1993.

[6] Calvo, A.B., and D.H. Marks, Location of health care facilities: an analytical approach, *Socio-Economic Planning Science*, vol.7, pp.407–422, 1973.

[7] Charnes, A., W.W. Cooper, and R.O. Ferguson, Optimal estimation of executive compensation by linear programming, *Management Science*, vol.1, pp.138–151, 1955.

[8] Charnes, A., and J. Storbeck, A goal programming model for the siting of multilevel EMS systems, *Socio-Economic Planning Science*, vol.14, no.4, pp.155–161, 1980.

[9] Church, R.L., and C. ReVelle, The maximal covering location problem, *Papers of the Regional Science Association*, vol.32, no.1, pp.101–118, 1974.

[10] Current, J., H. Min, and D.A. Schilling, Multiobjective analysis of facility location decisions, *European Journal of Operational Research,* vol.49, pp.295–307, 1990.

[11] Diwekar, U., *Introduction to Applied Optimization*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2003.

[12] Fayad, C., and S. Petrovic, A fuzzy genetic algorithm for real-world job shop scheduling, *Proc of Innovations in Applied Artificial Intelligence, Lecture Notes in Computer Science*, pp.524–533, 2005.

[13] Flynn, J., and S. Ratick, A multiobjective hierarchical covering model for the essential air service program, *Transportation Science*, vol.22, no.2, pp.139–147, 1988.

[14] Glover, F., Future paths for integer programming and links to artificial intelligence, *Computers and Operations Research*, vol.13, no.5, pp.533–549, 1986.

[15] Hakimi, S.L., Optimum locations of switching centres and the absolute centres and medians of a graph, *Operations Research*, vol.12, pp.450–459, 1964.

[16] Heller, M., J.L. Cohon, and C.S. ReVelle, The use of simulation in validating multiobjective EMS location model, *Annals of Operations Research*, vol.18, pp.303–322, 1989.

[17] Kanoun, I., H. Chabchoub, and B. Aouni, Goal programming model for fire and emergency service facilities site selection, *Proc of 7th International Conference on Multi-objective Programming and Goal Programming*, 2006.

[18] Kirkpatrick, S., C.D. Gellatt, and M.P. Vecchi, Optimization by simulated annealing, *Science,* vol.220, pp.671–680, 1993.

[19] Klir, G.J., U.S. Clair, and B. Yuan, *Fuzzy Set Theory: Foundations and Applications*, Prentice Hall, New Jersey, USA, 1997.

[20] Kwak, N.K., and M.J. Schniederjans, A goal programming model as an aid in facility location analysis, *Computers and Operations Research*, vol.12, no.2, pp.151–161, 1985.

[21] Marianov, V., and D. Serra, Hierarchical location-allocation models for congested systems, *European Journal of Operational Research*, vol.135, no.1, pp.195–208, 2001.

[22] Narasimhan, R., A fuzzy subset characterization of a site-selection problem, *Decision Sciences*, vol.10, no.4, pp.618–628, 1979.

[23] Narasimhan, R., Goal programming in a fuzzy environment, *Decision Sciences*, vol.11, pp.325–336, 1980.

[24] Narula, S.C., Hierarchical location-allocation problems: a classification scheme, *European Journal of Operational Research,* vol.15, pp.93–99, 1985.

[25] ReVelle, C.S., Facility siting and integer-friendly programming, *European Journal of Operational Research,* vol.65, pp.147–158, 1993.

[26] Sahin, G., and H. Süral, A review of hierarchical facility location models, *Computers and Operations Research*, vol.34, pp.2310–2331, 2007.

[27] Sinha, S.B., and S.V.C. Sastry, A goal programming model for facility location planning, *Socio-Economic Planning Science*, vol.21, no.4, pp.251–255, 1987.

[28] Tzeng, G.H., and Y.W. Chen, The optimal location of airport fire stations: a fuzzy multi-objective programming and revised genetic algorithm approach, *Transportation Planning and Technology*, vol.23, pp.37–55, 1999.

[29] Yang, L., B.F. Jones, and S.H. Yang, A fuzzy multi-objective programming for optimization of fire station locations through genetic algorithms, *European Journal of Operational Research*, vol.181, pp.903–915, 2007.

[30] Zadeh, L.A., Fuzzy sets, *Information and Control*, vol.8, pp.338–353, 1965.