Approximation Methods for Solving the Equitable Location Problem with Probabilistic Customer Behavior

J. Bagherinejad* & M. Omidbakhsh

Jafar Bagherinejad, Assistant Professor of Industrial Engineering, Faculty of Engineering and Technology, University of Alzahra, Tehran, Iran.
Maryam Omidbakhsh, MSc of Industrial Engineering, Faculty of Engineering and Technology, University of Alzahra, Vanak, Tehran, Iran

KEYWORDS
Facility location; Equitable load; Gravity model; Heuristic algorithm; Genetic algorithm; Integer programming

ABSTRACT
Location-allocation of facilities in service systems is an essential factor of their performance. One of the considerable situations which less addressed in the relevant literature is to balance service among customers in addition to minimize location-allocation costs. This is an important issue, especially in the public sector. Reviewing the recent researches in this field shows that most of them allocated demand customer to the closest facility. While, using probability rules to predict customer behavior when they select the desired facility is more appropriate. In this research, equitable facility location problem based on the gravity rule was investigated. The objective function has been defined as a combination of balancing and cost minimization, keeping in mind some system constraints. To estimate demand volume among facilities, utility function (attraction function) added to model as one constraint. The research problem is modeled as one mixed integer linear programming. Due to the model complexity, two heuristic and genetic algorithms have been developed and compared by exact solutions of small dimension problems. The results of numerical examples show the heuristic approach effectiveness with good-quality solutions in reasonable run time.


1. Introduction
Consider a location problem of some similar facilities in a geographical region. Suppose an incomplete capacity is costly, then all customers’ demand should be covered by the facilities in a fair way. The location problem in this case, is equivalent to find places that in addition to minimize costs of establishment and handling, minimize the demand volume in the busiest facility.

One of the important aspect of spatial interactions in location science is to predict customer behavior when they select a service system. Most models use proximity rule to estimate the demand flow, which is based on that each demand point selects only one facility by ignoring others. This assumption cannot correctly describe customer behavior because customers willing to receive service from the most attractive facility. So, we consider locating of similar facilities to meet customer needs so that demand allocating to the attractive facilities. This problem named as GBELP (Gravity-based Equitable Load Problem) and the objective of it is to minimize

* Corresponding author: Jafar Bagherinejad
Email: jbagheri@Alzahra.ac.ir
Paper first received Jan, 28, 2012, and in accepted form Sep, 14, 2012.
two measures namely, the maximum demand of each facility and the costs of location-allocation.

Literature review of research problems involves a probabilistic model for predicting a customer behavior and equity problem.

Employing gravity models in spatial interaction analysis is common, so we use this model for estimating people interactions accurately. Gravity model was first proposed by Reilly in 1931. Assuming that a customer is in one medium town near to two major cities, the probability of selecting one city has direct relation with city size and inversely proportion to distance. In this model, a decreasing function of distance was defined as the square of distance between customer and the city [1]. Huff [2,3] used Reily’s gravity model probability, to model the market share in competitive conditions and proved that a probability of selecting one shop by a customer has directly proportion to shop area and inverse ratio by some powers of its distance. When the power is infinitive, the customer selects the nearest shop with the probability of “1” [4].


Over the years, gravity rule has been used for solving problems in many areas such as geography (Lowe and Sen[15], Haynes and Fotheringham [16]), transportation planning (Evans [17], Erlander&Stewart [18]), marketing (Huff [20], Huff and Roland [19]) and particularly in location studies. For instance, Drezner and Drezner [20] and Eiselt and Marianov [21] applied it in hub location, Drezner and Drezner [5, 22] in median problem, O’kelly and Storbeck [24] in hierarchy location-allocation problem and Kokokaydin et.al [25] in competitive facility location.

Parallel to the development of this body of literature, a new field of research on location modeling was growing with equity objectives. Baron et.al [26] analyzed the problem of optimal location of a set of facilities in the presence of stochastic demand and congestion.

Galvao et.al [27] presented load balancing and capacity constraints in a hierarchical location model. Surana et.al [28] studied load balancing in dynamic structured peer-to-peer systems. Baatar and Wiseck [29] advance equity in multi-objective programming and strengthen the concept of Pareto efficiency by additionally requiring that the objective function be anonymous and satisfy the principle of transfers. Drezner and Drezner [30] investigated equity models in planar location.

In other papers [31], Researchers considered location model with two objectives: minimizing total distance traveled by customers and the variance of total demand attracted to each facility Baron et.al [32] considered the problem of locating facilities on the unit square so as to minimize the maximal demand faced by each facility subject to closest assignment and coverage constraints. Suzuki and Drezner [33] analyzed the minimum equitable radius location problem satisfying continuous area demand.

Berman et.al [34] studied network location such that the weights attracted to each facility will be as close as possible to one another. Puerto et.al [35] investigated extensive facility location problems with equity measures on networks. Objective functions measure conceptually related to the variability of the distribution of the distances from the demand points to a facility. Drezner et.al [36] locating facilities with equity considerations, namely, minimizing the Gini coefficient of the Lorenz curve based on service distances.

Properties of the Gini coefficient in the context of location analysis are investigated both for demand originating at points, and demand generated. Kostreva et.al [37] studied the concept of equitably efficient solutions (Equitable aggregations) to multiple criteria linear and non-linear optimization problems. Mesaa et.al [38] considered single facility location problems with equity measures, defined on networks. Galvao et al. [39] discuss practical issues in location problems of balancing loads of health-care facilities. Moreover, Kim and Kim focus on the problem of determining locations for long-term care facilities with the objective of balancing the numbers of patients assigned to the facilities.

Berman and Drezner and Baron et.al analyzes a location problem of similar facilities with a common service rate on one queue system [40, 41].

Based on existing literature, the related research about balancing workloads problem on the networks was presented by Berman et.al [34]. So, we selected it as the base paper of our study.

The contribution of our research is focusing on improving the solutions of this problem: location objective defined as the combination of decreasing establishment and transportation costs, besides fair distribution of demands among serving facilities. Client network supposed to have general form (not just tree). In demand allocation not only the proximity is important but also different desirability/undesirability factors be considered as important measures. So the probability of server selection defined as a function of the estimated attractiveness of each facility, distance, and current demand on it.

Heuristic and meta- heuristic approaches, in addition to the exact method, were used to solve the research problem. Findings show a very good efficiency according to the computational results.
2. GBELP\textsuperscript{2} Model

2-1. Customer Behavior

The flow rate between facilities $(g_{ij})$ estimated by the gravity rule, $g_{ij} = \frac{A_i}{F_{ij}}$. The expression often used for decreasing function of distance, is polynomial, $F_{ij} = d_{ij}^{-\alpha} (1 \leq \alpha \leq 3$, in most applications) or exponential, $F_{ij} = e^{-ad_{ij}}$. Distance effect will be different depending on the network structure. Erdog shows that results are not sensitive to distance function selection [42]. Huff proved distance effect by “product type” in practice. Distance function here defined as polynomial function $F_{ij} = d_{ij}^{-\alpha} + 1$, $\alpha \geq 0$, according to spatial position of point $i$ and facility $j$.

2-2. Notation

The required sets, parameters and decision variables are as Tab. 1:

| $I$ = {1, ..., n} | Set of demand points and also set of candidate locations for facilities. |
| $J$ = {1, ..., m} | Set of located facility. |
| $i$ | Demand points index. |
| $j$ | Facilities index. |
| $n$ | Number of demand points. |
| $m$ | Maximum number of facilities depends on the investment policy. $1 \leq m \leq n$. |
| $W_i$ | Node weight (demand rate) of point $i$. |
| $S_j$ | Current allocated demand to facility $j$. |
| $f_{ij}$ | The average capital cost to establish facility on the candidate node $j$, which may be different for each point; $f_{ij} > 0$. |
| $G$ | The estimated matrix of flow from demand point $i$ to facility $j$. |
| $C$ | The matrix of handling costs from demand point $i$ to facility $j$. |
| $D$ | The matrix of distances between network nodes. |
| $\alpha$ | The factor that the distance increased by it and determined empirically. |
| $M$ | A very large amount that can be considered as $M = Max_i (d_{ij})$. |
| $A$ | The $n \times n$ matrix of facility attractions, in each row further $A_{ij}$ is more interest facility at $j$ for demand point $i$. |
| $L_{\text{max}}$ | The maximum of allocated demand between all facilities. (The maximum load of facilities) |

2-3. Problem Formulation

The integer programming of GBELP is as follow:

$\text{Min } Z_1 = l_{\text{max}}$ (1)

$\text{Min } Z_2 = \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} d_{ij} y_{ij} + \sum_{i=1}^{n} f_i x_{ij}$ (2)

Subject to

$\sum_{j=1}^{m} w_{ij} \frac{A_j}{\sum_{j=1}^{m} A_j} - d_{ij} y_{ij} \leq L_{\text{max}} 1 \leq j \leq n$ (3)

$\sum_{j=1}^{m} x_{ij} \leq m$ (4)

$\sum_{j=1}^{m} y_{ij} = 1$ (5)

$y_{ij} \leq x_j$ (6)

$\sum_{j=1}^{m} y_{ij} d_{ij} + (M - d_{ij}) x_j \leq M$ (7)

$x_i \in \{0,1\}$ (8)

The objective function and constraints (3) ensure the minimax criterion, balancing the workloads. Equation (2) is the second objective function, in it the first expression is the establishment cost of facility in location $j$ and the second expression is handling cost of customer from demand point $I$ to facility $j$. In constraints (3), demands allocated to each facility by attraction function. The maximum value of allocated demand was in order as the maximum load ($L_{\text{max}}$).

Constraints (4) ensure that the number of facilities does not exceed from the specified value of $m$. Constraints (5), states that each node is assigned to one facility.

Constraints (6) ensure that the node $V_i$ cannot serve node $V_j$ unless there is a facility located at node $V_i$. Relations (8) and (9) show problem variables\textsuperscript{3} range. We prove that constraints (7) guarantee that each node is assigned to a closest facility. Let $J = \{ j | y_{i,j} = 1 \}$. For $j \notin J$ (7) is always true, because $x_j = 0$.

By (5) $y_{ij} = 0$ and therefore, the sum on the left hand side of (7) can be written as $\sum_{i=1}^{n} y_{ik} d_{ik} \leq d_{ij}$. If $y_{ik} = 1$, for $d_{ik} \geq \min (d_{ij})$, constraint (10) will be violated for $d_{ij} = \min (d_{ik})$. Therefore, $y_{ik}$ can be equals to 1 only for $d_{ik} = d_{ij}$ being the minimum distance.

If $U(x,y)$ and $V(x,y)$ were the efficiency and equity of one location pattern, respectively, in which $x,y$ is vector of decision variables, Then:

$U(x,y) = \max_{j} \{U_{\text{max}} (j) \}$ (11)

$V(x,y) = \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} d_{ij} y_{ij} + \sum_{i=1}^{n} f_i x_{ij}$ (12)

The above yardsticks not measured in the same unit. So, it is necessary to scale them, so that they can be converted to a common merit currency. We used

\textsuperscript{3}Gravity-based Equitable Load Problem ; GBELP
weighted metric method for this job. $U^*(x,y)$ and $V^*(x,y)$ are considered as the minimum of each yardstick in the absence of another. We can define such a function that could measure deviation from optimal solutions:

$$Z_1(x,y) = \frac{U(x,y) - U^*(x,y)}{U^*(x,y)}$$

(13)

$$Z_2(x,y) = \frac{V(x,y) - V^*(x,y)}{V^*(x,y)}$$

(14)

The converted efficiency and equity measures, weighted by an appropriate coefficient, are combined to formulate the model’s objective function ($Z(x,y)$):

$$Z(x,y) = \left( \frac{U(x,y) - U^*(x,y)}{U^*(x,y)} \right)^p + \left( 1 - \lambda \right) \left( \frac{V(x,y) - V^*(x,y)}{V^*(x,y)} \right)^q$$

(15)

$\lambda$ represents a weight assigned to each measure that between 0 (completely efficient) and 1 (completely equitable). Parameter p can take any value between 1 and $\infty$. The value of p determines emphasis on deviations from optimal solution, such that greater value has more emphasis on variation.

3. Solution Approaches

When all the attractiveness are equal and location-allocation cost is not considered, problem will change to ELP model which its complexity has been discussed by Berman et.al [34]. Let G = (V, E) be an undirected, connected graph with node set $V = \{v_i | i = 1, ..., n\}$ and edge set $E = \{e = (v_i, v_j) | i, j = 1, ..., n; i \neq j \}$. If all the distances between pairs of nodes are distinct, then for each subset M of m nodes there is only one feasible assignment. Therefore, by considering the $O\left(\binom{n^m}{m}\right)$ subsets of m nodes, problem ELP can be solved by complete enumeration in $O\left(\binom{n^m}{m}/(m-1)!\right)$ time on a general network and in particular, ELP is polynomially solvable for any fixed value of m. They observed that when m is variable and determine by $\binom{n^m}{m}$, the demand of each node ($w_i$), the distance between nodes ($d_{ij}$) and the unit cost of handling between them ($c_{ij}$), are the inputs.

ii. Shortest Paths

Dijkstra’s algorithm, conceived by Dutch computer scientist Edsger Dijkstra in 1956 is a graph search algorithm that solves the single-source shortest path problem for a graph with nonnegative edge path costs, producing a shortest path tree. This algorithm is often used in routing and as a subroutine in other graph algorithms.

Let the node at which we are starting be called the initial node. Let the distance of node Y is the distance from the initial node to Y. Dijkstra’s algorithm will assign some initial distance values and will try to improve them step by step.

1. Assign to every node a tentative distance value: set it to zero for our initial node and to infinity for all other nodes.
2. Mark all nodes except the initial node as unvisited.
3. Set the initial node as current. Create a set of the unvisited nodes called the unvisited set consisting of all the nodes except the initial node.
4. For the current node, consider all of its unvisited neighbors and calculate their tentative distances. For example, if the current node A is marked with a distance of 6, and the edge connecting it with a neighbor B has length 2, then the distance to B (through A) will be $6+2=8$. If this distance is less than the previously recorded distance, then overwrite that distance. Even though a neighbor has been examined, it is not marked as visited at this time, and it remains in the unvisited set.
5. When we are done considering all of the neighbors of the current node, mark the current node as visited and remove it from the unvisited set. A visited node will never be checked again; its distance recorded now is final and minimal.

We assume that the population volume is the most important factor in customer selection. Consider this as a measure, in addition to facility attraction, establishment cost and sum of its distance to other nodes.

So the gravity force of the facility in j, $T_j$, can be calculated by: $T_j = \frac{A_j}{d_j^p + 1}$ (16). $T_j$ is the establishment cost and $d_j$ is current population volume in node j. 1 is added to the denominator to prevent an infinite gravity force when $d_j = 0$. Final expression for determining gravity force of facility j is as follow:

$$C_{ij} = \frac{h_j/\delta_j}{d_j^p + 1} \cdot (d_{ij} + 1)^{-y}$$

(17).

i. Input

The node attractiveness for locating facility ($A_j$), establishment cost ($C_{ij}$), the maximum number of facilities (m), the demand of each node ($w_i$), the distance between nodes ($d_{ij}$) and the unit cost of handling between them ($c_{ij}$), are the inputs.
5. The next current node will be the node marked with
the lowest (tentative) distance in the unvisited set.
6. If the unvisited set is empty, then stop. The
algorithm has finished. Otherwise, set the unvisited
node marked with the smallest tentative distance as
the next “current node” and go back to step 3.

iii. Utility Fraction
To prioritize locations a Utility fraction,
\[ \text{c}(i) = \frac{d(i)}{t(i)} \quad i = 1, \ldots, n \]
calculated for n
nodes. \( c(i) \) and \( d(i) \) are the total distance and the total
handling cost of node \( i \) to the other points, respectively.

iv. Maximum Iteration (Stop Condition)
In this step, the iteration number determined based on
the \( m \) value. If \( m \leq 50 \), algorithm achieves to
acceptable solution by check all possible cases and in
one iteration \( (k=m, \text{MaxIt}=1) \). Otherwise, the interval
\([1,m]\) divided into four equal parts and for each of
these four values objective function calculated. We
considered two lowest values and assuming the
solution between these four objective functions. So this
interval divided into four parts, again. This procedure
repeated until a stop condition satisfied \( (k=\text{linspace}(1, m, 4), \text{MaxIt}= 2m-50) \).

v. Main Loop
The main loop computes the values of “Cost,
Maximum Load, Facility Locations, and Compound
Objective Function” and save them. Then it
investigates the combinations which more likely is
better solutions, depending on its utility fractions. Thus
the problem solved in a shorter run time than the exact
method.

vi. Output
The algorithm is stopped after a certain iteration
number \( (\text{MaxIt}) \) and finds the lowest objective
function, and returns its value with associated
maximum loads, cost and facility locations.

---

Fig. 1. The general structure of the Heuristic algorithm

Fig. 2. An example of GBELP chromosome
3.2. Genetic Algorithm (GA)

i. Basic Parameters and Inputs
The node attractiveness for locating facility \( (\mu_i) \), establishment cost \( (c_i) \), the maximum number of facilities \( (m) \), the demand of each node \( (d_i) \), the distance between nodes \( (d_{ij}) \) and the unit cost of handling between them \( (c_{ij}) \).

ii. Initial Population
Algorithm begins with an initial population of solutions. Each solution displayed as a “chromosome” and all chromosomes encoding by an appropriate code system. This code contains specific information:
1. Chromosome Location: A string of binary numbers with \( n \) dimension.
2. Fitness Function: The objective function of associated chromosome.

After the chromosome definition, a certain number of them \( (nPop) \) created randomly. nPop determined in a way that almost all solutions in a search space, have a chance to produce.

iii. Fitness Function
Solution evaluation done by a compounded function, \( Z(x,y) \). So less objective functions have a further fitness.

iv. Reproduction (Selection) Operator
Some chromosomes randomly select from population for reproduction. In the proposed GA, we used roulette wheel method for this selection. The probability of chromosome selection \( i \) calculated depending on its fitness, by equation (19):

\[
P_i = \frac{e^{-\beta Z_i}}{\sum_{j=1}^{n} e^{-\beta Z_j}}
\]

(19)

\( Z_j \) is the objective function of chromosome \( j \) and \( Z_{max} \) is the maximum of this function between all associated solution. \( \beta \) is a control parameter for chromosome selection (intensity parameter) and tuned in such a way that a half of chromosomes probability selection was 80%.

v. Crossover Operator
The crossover operator we used here is uniform. First two chromosomes selected and then one random chromosome with binary values created. One binary random distribution determined that each gens of children should pick from parents’ gen. So children created by equations (20) and (21).

\[
y_{1i} = \theta_{1} x_{1i} + (1 - \theta_{1}) x_{2i}
\]

(20)

\[
y_{2i} = \theta_{2} x_{2i} + (1 - \theta_{2}) x_{1i}
\]

(21)

vi. Mutation Operator
Mutation operator applies on chromosomes in order to prevent local optimum. We used the reversed mutation method in which one chromosome selected randomly and replaced by inverse value of it (Figure 4). This strategy prevented from too many changes in solutions.

vii. Merge Populations
By producing children, we should define a method to determine which member of the current population should be eliminated and which children should be replaced with them. This method effects on convergence of genetic algorithm [44]. All chromosomes in three populations sorted by their fitness and the bests of them selected as the chromosome number of the first population, for new generation.

ix. Stop Condition
Stop condition defined as achieving the maximum number of the generation production with the best population fit.

4. Computational Results

4-1. Numerical Examples
In this section the performance of the proposed algorithms is evaluated by using randomly generated instances with different problem sizes and other parameters. The basic parameters are generated as Tab.
2. Here the notation \( U(a,b) \) means a uniform distribution in the interval \([a,b]\) and \( Rnd(c,d) \) represents an integer random choice between \( c \) and \( d \). Fourteen problem sets are generated in order to evaluate the performance of the algorithms. So these instances of the problem are tested for basic parameters. We divided the instances into two categories: small scale (less than 40 customers), medium and large scale (more than 50 customers). To validate the model, random instances solved with GAMS 23.4, although this software is only able to solve the small size problems.

4-1. Solving Results and Comparison

The proposed algorithms are coded in MATLAB program and have been performed on a CPU of 2.2 GHZ and Windows 7. With the notations in Table 3, the results are given in Table 4.

<table>
<thead>
<tr>
<th>Tab. 2. Random Generated Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Parameter</strong></td>
</tr>
<tr>
<td>Maximum Number of facilities</td>
</tr>
<tr>
<td>Node Demand</td>
</tr>
<tr>
<td>Establishment Cost</td>
</tr>
<tr>
<td>Attractiveness</td>
</tr>
<tr>
<td>Distance between Locations</td>
</tr>
<tr>
<td>Handling Cost between Locations</td>
</tr>
</tbody>
</table>

**First we consider** \( m=n \)

To compare the proposed approaches some charts plotted based on Table 3. Computation Time (CT), Objective Function (Z) and Number of Function Evaluations (NFE) were compared with each other (Figures 5-7). We can conclude the following results:

1. Exact method computation time increased rapidly by problem dimension, while this increasing is very slow for GA and is imperceptible for heuristic algorithm. Moreover, with \( p \) parameter growth to infinite, computation time of the first approach increased but for approximation approaches this criterion is decreased (Figure 5). It seems logical because solving the model by adding one constraint in exact method is much simpler than solving the converted objective function of two algorithms in infinite case of \( p \) parameter.

2. The objective function of examples for \( p=1 \) is almost identical in three methods. But since increasing \( p \) in weighted metric methods leads to more realistic function for problems and heuristic algorithm has a near optimal solutions for \( p=\infty \), this method have priority over GA.

3. To measure the speed of convergence and the efficiency of approximation algorithms, we use NFE criterion besides CT and Z. This parameter states that one algorithm achieved to its best solution by evaluating how many solution between \( \sum_{i=1}^{n} (\frac{p}{i}) \) possible solutions.

5. Conclusion

In this research, a new type of location problem named the gravity-based equitable load problem was investigated. By considering the gravity model, realistic forecasting of customer behavior is provided, which also influence on the location of the facilities. The contribution of the paper to the literature are not only considering gravity model constraints but also integrating the equitable service and minimum location-allocation costs in the problem. So, an integer linear programming model is proposed and two approximation approaches are developed. The proposed algorithms were tested on different size of random instances and demonstrated that the heuristic algorithm provides good solutions in reasonable run time and with high-quality solutions for every dimension of the problems.

It is a very efficient algorithm with the average gap about 5.21% compared with the exact solutions. Furthermore, calculating the least number of function evaluation (NFE) is another advantage even for large problems with up to 60 demand points.

The algorithm is promising for practical applications to the public location problem. The following situations are suggested for future research in this field: The weight of the demand point can be divided among two or more facilities. Other equity measures such as minimizing the variance or the range of the total
demands can be considered. Developing GBELP in which model parameters are uncertain and have random distributions could enhance the validity of results in real-world applications.

We suggested branch and bound methods and also decomposition methods like Lagrangian relaxation for giving good lower bounds. The proposed multi objective problem could be solved by multi-objective meta-heuristic algorithms like NSGA-II, NRGA and MOPSO. Finally, the most important and obvious suggestion is applying the model for the real-world problems with field studies.

### Tab. 4. Performance of solutions in small, medium and large instances

<table>
<thead>
<tr>
<th>Instances</th>
<th>NFE</th>
<th>P</th>
<th>Size</th>
<th>NFE</th>
<th>RPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>GBELP-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>0.802</td>
<td>1.7820</td>
<td>0.1793</td>
<td>1.7954</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>3.244</td>
<td>1.8530</td>
<td>0.1347</td>
<td>1.8540</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7.0040</td>
<td>0.9308</td>
<td>0.1497</td>
<td>1.1778</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1.595</td>
<td>2.2420</td>
<td>0.6741</td>
<td>2.5219</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11.9410</td>
<td>1.1320</td>
<td>0.6633</td>
<td>2.2229</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7.2310</td>
<td>2.7250</td>
<td>1.9455</td>
<td>2.8966</td>
</tr>
<tr>
<td>GBELP-2</td>
<td>15</td>
<td>19.6570</td>
<td>2.7650</td>
<td>1.9561</td>
<td>2.7765</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>26.4530</td>
<td>1.8840</td>
<td>1.8342</td>
<td>1.9369</td>
</tr>
<tr>
<td></td>
<td></td>
<td>61.9310</td>
<td>2.5000</td>
<td>4.5821</td>
<td>3.0134</td>
</tr>
<tr>
<td>GBELP-3</td>
<td>20</td>
<td>304.4130</td>
<td>3.4720</td>
<td>9.0185</td>
<td>3.4795</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>407.6490</td>
<td>2.3300</td>
<td>8.6057</td>
<td>2.4267</td>
</tr>
<tr>
<td>GBELP-4</td>
<td>25</td>
<td>79.3380</td>
<td>3.2710</td>
<td>4.5028</td>
<td>3.2676</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>87.1450</td>
<td>2.0560</td>
<td>4.2783</td>
<td>2.0130</td>
</tr>
<tr>
<td></td>
<td></td>
<td>169.5670</td>
<td>3.3670</td>
<td>9.3033</td>
<td>3.4768</td>
</tr>
<tr>
<td>GBELP-5</td>
<td>30</td>
<td>304.4130</td>
<td>3.4720</td>
<td>9.0185</td>
<td>3.4795</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>407.6490</td>
<td>2.3300</td>
<td>8.6057</td>
<td>2.4267</td>
</tr>
<tr>
<td></td>
<td></td>
<td>520.2890</td>
<td>3.7880</td>
<td>16.9468</td>
<td>3.8158</td>
</tr>
<tr>
<td>GBELP-6</td>
<td>35</td>
<td>1123.6470</td>
<td>3.7630</td>
<td>16.4399</td>
<td>3.7811</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1567.5100</td>
<td>2.1740</td>
<td>15.8293</td>
<td>2.8257</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2773.2570</td>
<td>4.3860</td>
<td>28.4880</td>
<td>4.3845</td>
</tr>
<tr>
<td>GBELP-7</td>
<td>40</td>
<td>4173.1550</td>
<td>4.3810</td>
<td>27.6255</td>
<td>4.3983</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>5711.8420</td>
<td>2.1990</td>
<td>26.8662</td>
<td>2.4225</td>
</tr>
<tr>
<td></td>
<td></td>
<td>69.0550</td>
<td>4.5737</td>
<td>269.7991</td>
<td>3.4308</td>
</tr>
<tr>
<td>GBELP-8</td>
<td>40</td>
<td>1275.9485</td>
<td>4.4845</td>
<td>67.5949</td>
<td>4.8481</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1275.9485</td>
<td>4.4845</td>
<td>64.8997</td>
<td>2.6837</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>103.009</td>
<td>5.0872</td>
<td>997.401</td>
<td>3.8123</td>
</tr>
<tr>
<td>GBELP-9</td>
<td>50</td>
<td>1540.0128</td>
<td>4.8446</td>
<td>100.4128</td>
<td>4.8446</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>94.4799</td>
<td>2.9500</td>
<td>579.8032</td>
<td>3.6754</td>
</tr>
<tr>
<td>GBELP-10</td>
<td>50</td>
<td>1830.5060</td>
<td>3.1663</td>
<td>142.5060</td>
<td>3.1663</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>197.2155</td>
<td>5.9114</td>
<td>1972.9814</td>
<td>6.5438</td>
</tr>
<tr>
<td>GBELP-11</td>
<td>50</td>
<td>2145.3202</td>
<td>5.6229</td>
<td>188.3202</td>
<td>5.6229</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>184.9685</td>
<td>3.5017</td>
<td>1951.5244</td>
<td>4.0684</td>
</tr>
<tr>
<td>GBELP-12</td>
<td>50</td>
<td>2485.8304</td>
<td>3.2200</td>
<td>256.8304</td>
<td>3.2200</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>338.0442</td>
<td>6.8441</td>
<td>3829.4323</td>
<td>7.3126</td>
</tr>
<tr>
<td>GBELP-13</td>
<td>50</td>
<td>2850.3297</td>
<td>6.1272</td>
<td>329.3276</td>
<td>6.1272</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>331.3681</td>
<td>3.8367</td>
<td>3195.7567</td>
<td>4.0911</td>
</tr>
<tr>
<td>GBELP-14</td>
<td>50</td>
<td>3240.4210</td>
<td>6.3958</td>
<td>421.4200</td>
<td>6.3958</td>
</tr>
</tbody>
</table>

$\text{GAP}_{\text{ave}}$ indicator shows the average of total deviation in approximate value of the objective functions versus exact solution.
References


