Optimization
Iranian Journal of Optimization
Volume 13, Issue 3, 2021, 181-189
Research Paper

Online version is available on: www.ijo.rasht.iau.ir

Islamic Azad University
Rasht Branch
ISSN: 2588-5723
E-ISSN:2008-5427

# Use whale Algorithm and Neighborhood Search Metaheuristics with Fuzzy Values to Solve the Location Problem

Mehdi Fazli[1], Farzin Mosarres khiabani[1*] and Behrooz Daneshian[2]

[1] Department of Mathematics, Islamic Azad University, Tabriz Branch, Tabriz, Iran
[2] Department of Mathematics, Islamic Azad University, Central Tehran Branch, Tehran, Iran

**Revise Date**: 10 February 2022
**Accept Date**: 16 April 2022

## Abstract

In this paper, a facility location model with fuzzy value parameters based on the meta-heuristic method is investigated and solved. The proposed method and model uses fuzzy values to investigate and solve the problem of location allocation. The hypotheses of the problem in question are considered as fuzzy random variables and the capacity of each facility is assumed to be unlimited. This article covers a modern, nature-inspired method called the whale algorithm and the neighborhood search method. The proposed method and related algorithm are tested with practical optimization problems and modeling problems. To evaluate the efficiency and performance of the proposed method, we apply this method to our location models in which fuzzy coefficients are used. The results of numerical optimization show that the proposed method performs better than conventional methods.

**Keywords**:
Whale algorithm
Location problem
Fuzzy function
Neighborhood search algorithm
Meta heuristic

***Correspondence E-mail**: farrzln.modarres@gmail.com

## INTRODUCTION

Recently, meta-innovative techniques have been used to solve numerical problems and optimize complex real-world problems. Many of these methods are used by natural factors and phenomena and are valuable in solving various problems with very high dimensions. In particular, they can be used in specific topics related to different disciplines. The obvious and sometimes unique features of these methods are (i) Use of simple concepts and simplicity of hypotheses; (ii) No need for complex and extensive information; (iii) Don't get caught up in local optimism. Nature-inspired methods are used to solve problems related to Sari optimization and considering all the parameters of the problem. In such a way that all the details of the problem are considered to reach the optimal answer. In these methods, the initial search process begins with a group that has evolved randomly in the process and over subsequent generations. The bottom line is that great people are always coming together to form the next generation of improved people. The proposed method allows the population to produce a better generation during the period.

The first analysis and study of the dynamic location routing (LRP) dates back to the research conducted by Laporte and Dejax (1989)(Laporte & Dejax, 1989). They studied optimal and multiple programming cycles for LRPs, so they assumed each location and path in each cycle. They also analyzed the mental network profile of the problem. With this method, the problem of network optimization was solved with precise innovative approaches. Salhi and et al (1999)(Salhi & Nagy, 1999) assumed that problem warehouses and amenities were constant and unchanged along the planning route, but as the demand for new applicants changed, the traffic routes changed. It was also assumed that the capacity of each customer had not changed significantly. In their work, a number of paths and techniques were examined. Ambrosino and Scutella (2005)(Ambrosino & Scutella, 2005) investigated a multi-dimensional LRP using integer and static functional programming and practical software to answer real-world integer linear programming (ILP) problems. In these topics and in general, the problems related to location routing are discussed. The goal was to determine the capacity of each warehouse as a whole, a combination of customer service times and the available route from each warehouse to each number, to minimize the cost of the entire complex. In these methods, it is suggested that large examples of possible routing problems be solved with accurate and real values. A clustering algorithm based on Clark and Wright's algorithms was performed to receive acceptable and random hybridization solutions. Finally, the proposed method was reviewed in several sample sets and the results showed that the proposed method was better than the previous methods. Sambola et al (2012)(Albareda-Sambola, Fernández, & Nickel, 2012) presented the multi period routing problem with decoupled time scales. In their problem, there are certain limitations in which the system is forced to make important decisions in the field of routing and location. In addition, time scales are considered. They also assumed that the capacity and location of warehouses could be modified or expanded at the time chosen during planning. Given the variety and complexity of the model, they provide approximations based on vehicle replacement paths and warehouse changes and its ability to provide high quality solutions to a wide range of computational problems. In addition, the method of developing and combining these algorithms is often used for real applications and increasing the efficiency of methods. The whale algorithm is also used, which is inspired by nature, and the behavior of the predatory whale in pursuit of prey is modeled. This method simulates the movement of a whale spiral during an attack. And this is one of the methods used to optimize real-world problems.

## LITERATURE REVIEW

We see extensive literature on meta- heuristic techniques and their specific methods. And as you can see from the studies, most of these methods use an algorithm or a hybrid of them. In addition, different types of search algorithms can be used to improve the performance of the method in a given set. Or avoided problems such as stagnation in local optimization and loss of diversity in problem solving. In addition, there are techniques that hybridize the two techniques to reduce each other's weaknesses. Recently, a classification of different topics has been used to examine common applications between precision and metaheuristic approaches. To clarify each of these propositions accurately, the analysis of distinct types of compounds may be performed according to the introduced classifications. These specific categories of optimization problems can be divided into design and implementation problems. The following are options for problems in different departments:

• Relay, where a set of metaheuristics is applied one after another, or Teamwork, in which there are many parallel cooperating agents.

• Homogeneous, in which all hybrid methods use a particular heuristic, or heterogeneous, in which the system uses other different heuristics.

•Global, in which the method, algorithms search the entire research space, or partial, in which each algorithm seeks its own specific search space for the optimal answer.

•Specialist, which combines methods that solve different problems, or general, in which all methods of a problem optimize the problem.

•Specific, which solves only small and certain types of problems at much higher rates and low cost or versatility.

•Sequential, in which the methods work one by one and separately, and in parallel, in which each method and algorithm works at the same time as the other methods. A more complete study you can visit:

Goldberg and Holland (Goldberg, 1989) Koza (Koza, 1992), Simon (Simon, 2008), Alatas (Alatas, 2011), Kirkpatrick (Kirkpatrick, Gelatt, & Vecchi, 1983), Webster and Bernhard (Webster & Bernhard, 2003), Erol and Eksin (Erol & Eksin, 2006), Rashedi et al (Rashedi, Nezamabadi-Pour, & Saryazdi, 2009), Kaveh and Talatahari (Kaveh & Talatahari, 2010), Formato (Formato, 2007), Hatamlou (Hatamlou, 2013) Kaveh and Khayatazad (Kaveh & Khayatazad, 2012), Du and Zhuang (Du, Wu, & Zhuang, 2006), Moghaddam (Moghaddam, Moghaddam, & Cheriet, 2012), Shah-Hosseini (Shah-Hosseini, 2011), Gao et al (Gao, Sun, & Gen, 2008), Tavakkoli-Moghaddam et al (Tavakkoli-Moghaddam, Safaei, & Sassani, 2009), Drezner (Drezner, 2008), Jaszkiewicz and Kominek (Jaszkiewicz & Kominek, 2003) Lee et al(Lee, Su, Chuang, & Liu, 2008), Alba et al (Alba, Luque, & Troya, 2004), Chiu et al (Chiu, Chang, & Chang, 2007).

The proposed method is based on improving the performance of parts that did not have acceptable performance. The following sections describe the operation and features of the proposed method. The purpose of this study is to present an optimal method with good performance to find accurate solutions for application in optimization problems.

In this paper, a new meta-heuristic-based technique is presented. This proposal is based on the development of WA and the use of local search to solve location and routing problems. There are several reasons for developing this method:

• A review and then a detailed study of the parameters used in the method is performed. The problem population size and the specific parameters of each part of the technique are examined to obtain the best possible arrangement of the answer.

• A number of different measurement values of the problem are considered.

• This proposal is compared with new techniques with optimal performance and accuracy among other similar methods.

• This technique uses fuzzy data to achieve acceptable performance compared to similar statistical tests and real problems.

The rest of the discussion is as follows. In Section 3, after a brief introduction to WA, the proposed method for develop this algorithm. In Section 4, we discuss the location problem, and we test the performance of the proposed method on some numerical problems at different scales and with numerical tests, we show the efficiency of this method.

## MODEL FORMULATION

### whale algorithm and variable neighborhood search

An important point about Reino prey whales is their special methods of trapping their prey. These instinctive behaviors are referred to as their eating patterns, which are inspired by nature(Watkins & Schevill, 1979). Nevertheless, Goldbogen et al (Goldbogen et al., 2013)examined this conduct using tag sensors. It should be noted that this method of feeding pure bubbles is a special treatment that is seen only in some whales. To optimize various problems in mathematical models, the spiral hunting method is used, which is the basis of this research.

In the WA method, it is assumed that: The current candidate is the optimal or near-optimal solution. After determining the best option, other problem factors are added and their place in the search engine is explored. The following equations show this process:

$$\vec{F} = \left|\vec{M} \cdot \vec{T^*}(l) - \vec{T}(l)\right| \quad (1)$$

$$\vec{T}(l+1) = \vec{T^*}(l) - \vec{N} \cdot \vec{F} \quad (2)$$

where $l$ indicates the current iteration, $T^*$ is the position vector of the best solution obtained so far, $\vec{N}$ and $\vec{M}$ are the coefficient vectors, | | is the absolute value, $\vec{T}$ is the position vector, and · is an element-by-element multiplication. In order to get the best answer or optimal, $T^*$ must be updated to repeat the problem. The vectors $\vec{N}$ and $\vec{M}$ are represented as follows:

$$\vec{N} = 2\vec{b} \cdot \vec{s} - \vec{b} \quad (3)$$

$$\vec{M} = 2 \cdot \vec{s} \quad (4)$$

Where $\vec{b}$ is linearly reduced to 0 over the course of repeats (in both exploitation phases and exploration) and $\vec{s}$ is a random vector in [0,1]. The same concept can be developed in the n-dimensional problem search space, and search agents are over-searching around the optimal solution. According to what we mentioned in this section, predatory whales attack their designated targets. This particular method of mathematical optimization is as follows:

Two specific methods have been investigated to model the net bubble behavior of humpback whales in the face of predation:

1. Specific method of contractile encirclement: This conduct is achieved by decreasing the amount of $\vec{b}$ in equation. The fluctuation domain of $\vec{N}$ is also changed by $\vec{b}$.

Nevertheless, $\vec{N}$ is a random value in the interval $[-z, z]$ where z decreases to 0 during iterations. By hypothetical random values for $\vec{N}$ in $[-1,1]$, the new position of a search agent can be defined anywhere in between the main position of the agent and the position of the best current agent. Results show the possible locations from $(T, Y)$ towards $(T^*, Y^*)$ that can be attained by $0 \leq C \leq 1$ in a 2-dimensional space.

2. Spiral updating location: This approach first calculates the distance between the whale placed at $(T, Y)$ and prey placed at $(T^*, Y^*)$. In the next step, in order for the spiral motion to mimic the shape of the humpback whales, the spiral equation between the location of the whale and the prey is created as follows:

$$\vec{T}(l+1) = \acute{F} \cdot e^{rt} \cdot \cos(2\pi t) + \vec{T^*}(l) \quad (5)$$

Where $\vec{\acute{F}} = \left|\vec{T^*}(k) - \vec{T}(k)\right|$ and shows the distance of the $i$th whale to the prey, $r$ is a constant value for defining the shape of the logarithmic spiral, is an element-by-element multiplication, and t is a assumptive number in $[-1,1]$. Note that these whales move around a hunt in a limited circle along the spiral path. We assume that the probability of choosing between the helical model or the siege mechanism to update the whale position during optimization is 50%, to model this behavior. The model is described below:

$$\vec{T}(l+1) = \begin{cases} \vec{T^*}(l) - \vec{N}.\vec{F} & \text{if } p < 0.5 \\ \acute{F} \cdot e^{rt} \cdot \cos(2\pi t) + \vec{T^*}(l) & \text{if } p > 0.5 \end{cases} \quad (6)$$

Here $p$ is an assumptive number in [0,1]. Humpback whales also search for prey at random, and the math search model is below:

$$\vec{F} = \left|\vec{M}.\overrightarrow{T_{rand}} - \vec{T}\right| \qquad (7)$$

$$\vec{T}(l+1) = \overrightarrow{T_{rand}} - \vec{N}.\vec{F} \qquad (8)$$

Usually in the initial repetitions of the whale method, significant parts of the problem space may not be searchable. Therefore, implementing WA on existing population members may increase execution time without achieving significant improvement.So, the proposed method is to use (variable neighborhood search) VNS in the WA method, which is less time to achieve the desired result. have. Then, as we recognize the optimal solution space confidence interval, we slowly increase the likelihood of using VNS, called $P_{VNS}$, on the population. Since the issue of extending the WA method was considered, we expanded the developed method called WAVNS. In this method, this method uses the following update rule:

$$P_{VNS} \leftarrow \beta_{VNS} P_{VNS} \qquad (9)$$

where $\beta_{VNS} > 1$ $(if\ P_{VNS} > 1, then\ we\ set\ P_{VNS} = 1)$

**Algorithm WA with neighborhood search**

```
Exploration model implemented in WA ( T *is the position vector of the best solution obtained so far).
Initialize the Whales collection Ti(i = 1, 2, ..., n)
 Compute the fitness of each search factor
T*=the best search factors
While (l< maximum value of iterations)
      for each search factor
     Update b, N, M, t, and p
          if1(p<0.5)
               if2(|N|< 1)
               Update the location of the current search factor by Eq.
               else if2(|N| ≥ 1 )
                    Select a assumptive search factor (Trand)
                  Update the location of the current search factor by Eq.Error! Reference source not
                  found.
               Perform the VNS on the best individual of Ptemp (if there are several, select one randomly)
               with probability PVNS.
               end if2
         elseif1(p ≥ 0.5)
                  Update the location of the current search by Eq.
         end if1
      end for
      Check if any search factor goes beyond the search space and amend it
      Compute the fitness of each search factor
      Update T* if there is an optimal solution
      l=l+1
end while
return T*
```

## NUMERICAL EXPERIMENTS

This section describes a specific case of location and allocation problems. As you can see, the proposed method and algorithm will be very effective. In numerical optimization programs, the assumptions of a practical and real problem, such as the exact amount required and the results, are often incorrect. In fact, to avoid this problem, we examine the location of the facility with fuzzy values for these hypotheses and provide a degree of freedom for the decision maker that allows for uncertainty in the input data and assumptions. A special natural technique for describing unreliable data is the use of variables and fuzzy data. Hence, here we describe a location allocation formula, i.e. fuzzy station location allocation, with setup costs and associated applicants. To test the proposed algorithm, we created some environmental problems to test the large-scale fuzzy location problem (FLP). Similar to the mathematical formula of the place problem as an integer programming question (16), the FLP formula by:

$$max = Z(x, y) = \mathcal{R}(\sum_{i\in I}\sum_{j\in J_i^+} \chi_{ij}\mu_{ij} f(c_{ij})\tilde{d}_j - \sum_{i\in I}\tilde{f}_i y_i)$$

$$= \sum_{i\in I}\sum_{j\in J_i^+} \chi_{ij}\mu_{ij} f(c_{ij})\mathcal{R}(\widetilde{d}_j) - \sum_{i\in I}\mathcal{R}(\tilde{f}_i) y_i$$

$$\text{s,t,} \quad \sum_{i\in I_j^+} \chi_{ij} \le 1 \quad \forall j \in J \qquad (10)$$

$$\sum_{i\in J_i^+} \chi_{ij} \le |J_i^+| y_i \quad \forall i \in I \qquad (11)$$

$$k_{min} \le \sum_{i\in I} y_i \le k_{max} \qquad (12)$$

$$\chi_{ij} \in \{0,1\} \quad \forall i \in I,\ j \in J \qquad (13)$$

$$y_i \in \{0,1\} \quad \forall i \in I, \qquad (14)$$

Consider that the limitations (4.2) guaranty that when node $i \in I$ is selected as a point or terminal ($y_i = 1$), next it can service and cover all the node sin $J + i$, while the limitation (4.3) controls the numeral of the needed terminals.

Often, all node locations and their terminals in the range [-10, 10] × [-10, 10] were assumed and selected with uniform distribution. Consider that $m = |I|$, $n = |J|$ For each node of the problem, $j \in J, \tilde{d}j = (a_j^L, a_j^L + t_j, \alpha_j, \beta_j), j = 1,\ldots, n,$ is a trapezoidal fuzzy value, where $\alpha_j,\beta_j \sim U[10, 50], a_j^L \sim U[500, 2500]$, and $tj \sim U[0, 100]$. Here considered the function of $J_i^+(i = 1, \ldots. m)$ as follows:

$$\mu_{J_i^+}(j) = \mu_{ij} = \begin{cases} 1, & c_{ij} \le r \\ 1 + \dfrac{r - c_{ij}}{d_r}, & r \le c_{ij} \le r + d_r, \\ 0, & c_{ij} > r + d_r \end{cases} \qquad (15)$$

Shrinking encircling mechanism is achieved by decreasing the value of $\vec{b}$ in Equation. Note that the fluctuation range of $\vec{N}$ is also decreased by $\vec{b}$.

Additionally, in all runs, we set $r = 1$ and $d_r = 0.1$ in (4.6)

In numerical problems was set m = 250,500,750,1000 and n = 4m. We conducted our calculations in the MATLAB 9.0 programming setting on a computer, Intel(R) Core (TM)i7-7500U CPU@ 2.90 GHz, with 12 GB of RAM.

We developed the whale method with the help of local search and fuzzy values. Therefore, the performance of all similar methods is observed using the same execution time. The names and specifications of the test problems and the processing time required for the methods are given in Table 1.

In each applied problem, the numerical result of the number 1 is assigned to find the best solution, ie the solution and the answer that has the least relative error. And the solution or solution that has the most relative error, ie the worst solution, will take the number 0, and the rest of the numerical answers, depending on how close it is to the best solution, take the values from 0 to 1. In simpler terms, if the maximum relative error value obtained from all methods in the particular problem j is denoted by ej, and the relative error obtained for the corresponding method and algorithm i in the problem j is denoted by $e_{ij}$, we consider $1 - \frac{e_{ij}}{e_j}$ as the numerical result for algorithm $i$ on problem $j$. Table 2 shows the numerical results obtained.

To test and demonstrate WAVNS's competitiveness in achieving appropriate and optimal quality solutions, we implemented and tested other algorithms on all applied experimental problems using higher values for time constraints. We set all other related methods and algorithms to the2 $\times T_{WA}$ time limit. In the next step, for a certain method in the test problem, we considered the score as the following formula.

$$s_i(\text{alg}) = \begin{cases} 2, & \text{alg could find a better solution than WAVNS} \\ 1, & \text{alg could find the WAVNS solution,} \\ 0, & \text{O.W.} \end{cases} \qquad (16)$$

Numerical results demonstrate that WAVNS, HGAVNS (Hybrid genetic and variable neighborhood search) and MVNS methods have found the best solution in 89.64%, 6.92% and 3.45%, but other methods cannot reach the best solution. Compared to GA, MSA and MVNS methods, only MVNS and GA achieved the best solution in 70% and 30% of cases, respectively, and HGAVNS was better than NHGASA, Lin and HGASA, and MSA (multistart simulated annealing algorithm) was the worst.

Fig. 1 shows the efficiency of the proposed method and this value has been compared with other similar methods.

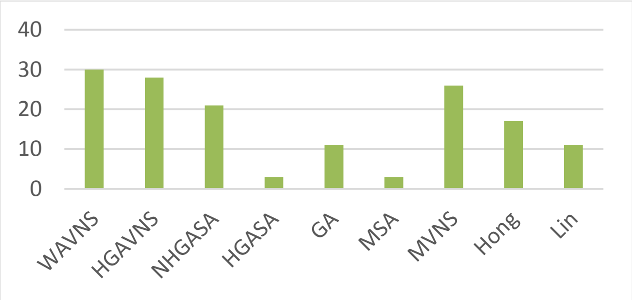


Fig.1. Shows the scores of the methods calculated by (16)

Table 1: Test problems specifications.

| Problem | Category | n | $k_{min}$ | $k_{max}$ | Time (s) |
|---|---|---|---|---|---|
| 1 | 1 | 250 | 25 | 50 | 0.1036 |
| 2 | 2 | 250 | 25 | 50 | 0.163339 |
| 3 | 3 | 250 | 25 | 50 | 0.101057 |
| 4 | 1 | 250 | 50 | 100 | 0.093915 |
| 5 | 2 | 250 | 50 | 100 | 0.093915 |
| 6 | 3 | 250 | 50 | 100 | 0.093915 |
| 7 | 1 | 250 | 100 | 125 | 0.107203 |
| 8 | 2 | 250 | 100 | 125 | 0.106835 |
| 9 | 3 | 250 | 100 | 125 | 0.101188 |
| 10 | 1 | 500 | 50 | 100 | 0.206169 |
| 11 | 2 | 500 | 50 | 100 | 0.141496 |
| 12 | 3 | 500 | 50 | 100 | 0.135905 |
| 13 | 1 | 500 | 100 | 200 | 0.116595 |
| 14 | 2 | 500 | 100 | 200 | 0.144836 |
| 15 | 3 | 500 | 100 | 200 | 0.147267 |
| 16 | 1 | 500 | 200 | 250 | 0.127699 |
| 17 | 2 | 500 | 200 | 250 | 0.133205 |
| 18 | 3 | 500 | 200 | 250 | 0.139434 |
| 19 | 1 | 750 | 75 | 150 | 0.160926 |
| 20 | 2 | 750 | 75 | 150 | 0.153569 |
| 21 | 3 | 750 | 75 | 150 | 0.148119 |
| 22 | 1 | 750 | 150 | 300 | 0.137819 |
| 23 | 2 | 750 | 150 | 300 | 0.146553 |
| 24 | 3 | 750 | 150 | 300 | 0.150449 |
| 25 | 1 | 750 | 300 | 375 | 0.140989 |
| 26 | 2 | 750 | 300 | 375 | 0.145201 |
| 27 | 3 | 750 | 300 | 375 | 0.147101 |
| 28 | 1 | 750 | 100 | 200 | 0.169967 |
| 29 | 2 | 1000 | 100 | 200 | 0.165532 |
| 30 | 3 | 1000 | 100 | 200 | 0.166014 |
| 31 | 1 | 1000 | 200 | 400 | 0.168161 |
| 32 | 2 | 1000 | 200 | 400 | 0.171869 |
| 33 | 3 | 1000 | 200 | 400 | 0.171738 |
| 34 | 1 | 1000 | 400 | 500 | 0.163274 |
| 35 | 2 | 1000 | 400 | 500 | 0.161609 |
| 36 | 3 | 1000 | 400 | 500 | 0.158127 |

Table 2: Numerical results (performances)

| Problem | WAVNS | HGAVNS | NHGASA | HGASA | GA | MSA | MVNS | Hong | Lin |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.8262 | 0.6504 | 0.8022 | 0.0682 | 0.8128 | 0 | 0.0217 | 0.7432 | 0.6185 |
| 2 | 0.9755 | 0.8702 | 0.8553 | 0 | 0.7484 | 0.0255 | 0.8231 | 0.8609 | 0.8376 |
| 3 | 0.9321 | 0.9204 | 0.8544 | 0.1847 | 0.571 | ۰.0۰۰۰ | 0.8423 | 0.6752 | 0.6978 |
| 4 | 0.9832 | 0.8314 | 0.9426 | 0.1151 | 0.9628 | 0 | 0.6881 | 0.8921 | 0.8534 |
| 5 | 0.9537 | 0.9678 | 0.8413 | 0.0458 | 0.6229 | 0 | 0.9766 | 0.7904 | 0.6903 |
| 6 | 0.9429 | 0.9131 | 0.8234 | 0.0481 | 0.3336 | 0 | 0.9415 | 0.8469 | 0.7465 |
| 7 | 0.9641 | 0.739 | 0.8243 | 0.1544 | 0.7829 | 0 | 0.7888 | 0.8123 | 0.7424 |
| 8 | 0.9728 | 0.8423 | 0.8147 | 0.0683 | 0.5075 | 0 | 0.8623 | 0.7967 | 0.7321 |
| 9 | 0.9514 | 0.8629 | 0.8135 | 0.1102 | 0.3095 | 0 | 0.8637 | 0.7836 | 0.7023 |
| 10 | 0.9347 | 0.4568 | 0.7848 | 0.0341 | 0.7871 | 0 | 0.1461 | 0.7601 | 0.7287 |
| 11 | 0.9218 | 0.0016 | 0.2608 | 0.2121 | 0 | 0.0129 | 0.2338 | 0.1918 | 0.2004 |
| 12 | 0.8853 | 0.7518 | 0.7382 | 0.0301 | 0.343 | 0 | 0.7999 | 0.7534 | 0.8625 |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 13 | 0.9375 | 0.8095 | 0.8406 | 0.347 | 0.8829 | 0 | 0.8015 | 0.8666 | 0.8209 |
| 14 | 0.9632 | 0.8875 | 0.7766 | 0 | 0.4512 | 0.1079 | 0.9191 | 0.6954 | 0.6217 |
| 15 | 0.9844 | 0.9432 | 0.7493 | 0.1025 | 0.2385 | 0 | 0.9175 | 0.6578 | 0.6625 |
| 16 | 0.9729 | 0.8348 | 0.7745 | 0.0229 | 0.6362 | 0 | 0.8332 | 0.651 | 0.6019 |
| 17 | 0.9587 | 0.8687 | 0.7407 | 0.0465 | 0.2544 | 0 | 0.8836 | 0.5025 | 0.4867 |
| 18 | 0.9712 | 0.9062 | 0.7354 | 0.0548 | 0.214 | 0 | 0.9161 | 0.5298 | 0.4806 |
| 19 | 0.9486 | 0 | 0.9334 | 0.7336 | 0.9204 | 0.7177 | 0.8166 | 0.9253 | 0.9237 |
| 20 | 0.7994 | 0.6902 | 0.717 | 0.0386 | 0.7845 | 0 | 0.6549 | 0.6327 | 0.6416 |
| 21 | 0.9861 | 0.7553 | 0.6508 | 0 | 0.2921 | 0.0127 | 0.7928 | 0.5907 | 0.3718 |
| 22 | 0.9943 | 0.895 | 0.8077 | 0.047 | 0.8171 | 0 | 0.9103 | 0.7583 | 0.7509 |
| 23 | 0.9872 | 0.9662 | 0.5348 | 0 | 0.0116 | 0.0168 | 0.8975 | 0.3287 | 0.2953 |
| 24 | 0.9328 | 1 | 0.6229 | 0.0404 | 0.1464 | 0 | 0.9109 | 0.5698 | 0.5512 |
| 25 | 0.9239 | 0.871 | 0.7323 | 0.0515 | 0.5605 | 0 | 0.8932 | 0.7012 | 0.619 |
| 26 | 0.9873 | 0.9035 | 0.6574 | 0.0418 | 0.2405 | 0 | 0.9043 | 0.6423 | 0.583 |
| 27 | 0.9956 | 0.9168 | 0.6626 | 0.0396 | 0.1508 | 0 | 0.9255 | 0.6217 | 0.6021 |
| 28 | 0.9327 | 0.8758 | 0.9164 | 0.5427 | 0.9204 | 0 | 0.8536 | 0.8709 | 0.8841 |
| 29 | 0.7985 | 0.5291 | 0.7495 | 0.0254 | 0.6807 | 0 | 0.4136 | 0.709 | 0.6823 |
| 30 | 0.7838 | 0.494 | 0.7563 | 0.0328 | 0.6597 | 0 | 0.4127 | 0.7713 | 0.7384 |
| 31 | 0.9871 | 0.9111 | 0.7315 | 0.0363 | 0.7752 | 0 | 0.985 | 0.7465 | 0.6982 |
| 32 | 0.9126 | 0.9926 | 0.7203 | 0.0246 | 0.8017 | 0 | 0.8453 | 0.781 | 0.7267 |
| 33 | 0.9877 | 0.9089 | 0.747 | 0.0134 | 0.8123 | 0 | 0.9583 | 0.7606 | 0.7724 |
| 34 | 0.9749 | 0.9018 | 0.6636 | 0.0608 | 0.4948 | 0 | 0.8785 | 0.5763 | 0.6081 |
| 35 | 0.9623 | 0.8663 | 0.6061 | 0.0296 | 0.4784 | 0 | 0.8911 | 0.4792 | 0.421 |
| 36 | 0.9539 | 0.8629 | 0.6232 | 0.0212 | 0.4604 | 0 | 0.8647 | 0.5691 | 0.5258 |
| Average | 0.9411 | 0.7788 | 0.7445 | 0.0951 | 0.5407 | 0.0248 | 0.7796 | 0.6901 | 0.6522 |

## CONCLUSIONS

In this research paper, we develop and implement a meta-heuristic method to solve a location problem that uses fuzzy values. We also compare it with recently implemented methods and algorithms to prove the efficiency and effectiveness of our technique. A model with fuzzy values also had a fuzzy number of node-related applicants, with lower limits and a predefined limit for the number of stations. We tested the proposed method and related algorithm on various fuzzy station problems with several random variables in which the cost of the fuzzy value system is considered. The fuzzy target value in this problem was converted to an explicit value using a ranking equation. As can be seen, numerical experiments on real-size application problems have a desirable and acceptable effectiveness.

## REFERENCES


Alatas, B. (2011). ACROA: Artificial Chemical Reaction Optimization Algorithm for global optimization. Expert Syst. Appl., 38(10), 13170–13180. doi:10.1016/j.eswa.2011.04.126

Alba, E., Luque, G., & Troya, J. M. (2004). Parallel LAN/WAN heuristics for optimization. Parallel Computing, 30(5-6), 611-628.

Albareda-Sambola, M., Fernández, E., & Nickel, S. (2012). Multiperiod location-routing with decoupled time scales. European journal of operational research, 217(2), 248-258.

Ambrosino, D., & Scutella, M. G. (2005). Distribution network design: New problems and related models. European journal of operational research, 165(3), 610-624.

Chiu, Y. C., Chang, L. C., & Chang, F. J. (2007). Using a hybrid genetic algorithm–simulated annealing algorithm for fuzzy programming of

reservoir operation. Hydrological Processes, 21(23), 3162-3172.
Drezner, Z. (2008). Extensive experiments with hybrid genetic algorithms for the solution of the quadratic assignment problem. Computers & Operations Research, 35(3), 717-736.
Du, H., Wu, X., & Zhuang, J. (2006). Small-world optimization algorithm for function optimization. Paper presented at the International Conference on Natural Computation.
Erol, O. K., & Eksin, I. (2006). A new optimization method: big bang–big crunch. Advances in Engineering Software, 37(2), 106-111.
Formato, R. (2007). Central force optimization: A new metaheuristic with applications in applied electromagnetics. Progress in electromagnetics research. PIER 77, 425–491. In.
Gao, J., Sun, L., & Gen, M. (2008). A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems. Computers & Operations Research, 35(9), 2892-2907. doi:https://doi.org/10.1016/j.cor.2007.01.001
Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization and Machine Learning: Addison-Wesley Longman Publishing Co., Inc.
Goldbogen, J. A., Friedlaender, A. S., Calambokidis, J., Mckenna, M. F., Simon, M., & Nowacek, D. P. (2013). Integrative approaches to the study of baleen whale diving behavior, feeding performance, and foraging ecology. BioScience, 63(2), 90-100.
Hatamlou, A. (2013). Black hole: A new heuristic optimization approach for data clustering. Information sciences, 222, 175-184.
Jaszkiewicz, A., & Kominek, P. (2003). Genetic local search with distance preserving recombination operator for a vehicle routing problem. European journal of operational research, 151(2), 352-364.
Kaveh, A., & Khayatazad, M. (2012). A new meta-heuristic method: ray optimization. Computers & structures, 112, 283-294.
Kaveh, A., & Talatahari, S. (2010). A novel heuristic optimization method: charged system search. Acta Mechanica, 213(3-4), 267-289.
Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. science, 220(4598), 671-680.
Koza, J. R. (1992). Genetic Programming II, Automatic Discovery of Reusable Subprograms: MIT Press, Cambridge, MA.
Laporte, G., & Dejax, P. J. (1989). Dynamic location-routeing problems. Journal of the Operational Research Society, 40(5), 471-482.
Lee, Z.-J., Su, S.-F., Chuang, C.-C., & Liu, K.-H. (2008). Genetic algorithm with ant colony optimization (GA-ACO) for multiple sequence alignment. Applied Soft Computing, 8(1), 55-78.
Moghaddam, F. F., Moghaddam, R. F., & Cheriet, M. (2012). Curved space optimization: a random search based on general relativity theory. arXiv preprint arXiv:1208.2214.
Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). GSA: a gravitational search algorithm. Information sciences, 179(13), 2232-2248.
Salhi, S., & Nagy, G. (1999). Consistency and robustness in location-routing. Studies in Locational Analysis(13), 3-19.
Shah-Hosseini, H. (2011). Principal components analysis by the galaxy-based search algorithm: a novel metaheuristic for continuous optimisation. International Journal of Computational Science and Engineering, 6(1-2), 132-140.
Simon, D. (2008). Biogeography-Based Optimization. Trans. Evol. Comp, 12(6), 702–713. doi:10.1109/tevc.2008.919004
Tavakkoli-Moghaddam, R., Safaei, N., & Sassani, F. (2009). A memetic algorithm for the flexible flow line scheduling problem with processor blocking. Computers & Operations Research, 36(2), 402-414.
Watkins, W. A., & Schevill, W. E. (1979). Aerial observation of feeding behavior in four baleen whales: Eubalaena glacialis, Balaenoptera borealis, Megaptera novaeangliae, and Balaenoptera physalus. Journal of Mammalogy, 60(1), 155-163.
Webster, B., & Bernhard, P. J. (2003). A local search optimization algorithm based on natural principles of gravitation. Retrieved from.