IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 8, No. 5, October, 2021, pp. 763-768
Journal homepage: http://iieta.org/journals/mmep

Hybrid Whale and Genetic Algorithms with Fuzzy Values to Solve the Location Problem

Mehdi Fazli[1], Farzin M. Khiabani[1*], Behrouz Daneshian[2]

[1] Department of Mathematics, Islamic Azad University, Tabriz Branch, Tabriz 5157944533, Iran
[2] Department of Mathematics, Islamic Azad University, Central Tehran Branch, Tehran 1955847781, Iran

Corresponding Author Email: f.modarres@iaut.ac.ir

https://doi.org/10.18280/mmep.080511

Received: 23 February 2021
Accepted: 26 May 2021

Keywords:
fuzzy function, generic algorithm, location problem, whale algorithm

ABSTRACT

In this paper, a facility location model with fuzzy values parameters based on the hybrid meta-heuristic method is investigated. The proposed model uses fuzzy values to solve the installation problem. Problem hypotheses are considered fuzzy random variables, and the capacity of each facility is unlimited. This paper combines a modern nature-inspired procedure called the Whale Algorithm (WA) with genetic methods. WA and Genetic Algorithm (GA) have been tested with scientific optimization problems and modeling problems. To evaluate the performance of the proposed methods, we apply the methods to our spatial models in which fuzzy coefficients are used. The results of numerical optimization show that the proposed combined method performs better than conventional methods.

1. INTRODUCTION

Lately, meta-heuristic techniques have been used to solve numerical problems and optimize real-world problems. Many of these methods are used by natural phenomena, and they are valuable in solving issues of very high dimensions. In general, they can be used in different issues related to various disciplines. Clear features of these methods are: (i) the use of simple concepts and simplicity; (ii) no need for abundant information; (iii) Don't get caught up in local optimism. To solve the optimization problems, nature-inspired methods are used, which try to solve the problem by considering many parameters, in such a way that it uses evolution-based methods. In these methods, the search process begins with a group that has evolved randomly over the next generations. The final subject of these methods is that the premium individuals are ever combined to form the forthcoming generation of individuals. Presented method allows the population to be better over the period of descendant.

The first study on dynamical Location Routing Problem (LRP) dates back to the research conducted by Laporte and Dejax [1]. They studied together multiplex programming cycles for LRPs, thus putting up each location and route in each cycle. They also investigated a mental network profile of the problem. The problem of network optimization was solved with detailed heuristic approaches. Salhi and Nagy [2] assumed that problem warehouses and convenience were fixed along the planning route, but as the demand for new applicants changed, the vehicle 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 [3] investigated a multi-dimensional LRP using integer and static functional programming and practical software to answer real-world integer linear programming (ILP) problems. In general, problems with location routing are discussed. The goal was to determine the capacity of a warehouse as a whole, the combination of customer service times and the route available from each warehouse to each issue, to minimize the total cost of the collection. It is proposed to solve large examples of likely routing problems with 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 investigated in several sample sets, and the results showed that the previous methods were better. Albareda-Sambola et al. [4] presented the multi period routing problem with decoupled time scales. Their problem is specified by specific constraints in which routing and location decisions are made on the time scales presented. They also assumed that warehouses could be modified or expanded at the time selected during planning. Given the variety and complexity of the model, they provide approximations based on vehicle replacement routes and warehouse changes and its ability to provide good quality solutions to a wide range of computational problems. Genetic algorithms (GA) [5] are one of the most universal evolution-inspired techniques, This program is very applied and follows the evolution of Darwin. In addition, the development method of these algorithms is often used to increase the efficiency of the methods. Also, the whale algorithm is derived from the nature of the predatory whale in pursuit of prey, which is inspired by nature. Spiral motion simulates the mechanism of a predatory whale attack. The whale algorithm is one of the methods used to optimize real problems.

2. LITERATURE REVIEW

The literature on meta-heuristic techniques is very extensive. As can be seen, most of the proposed techniques use a meta-heuristic or hybrid algorithm. In addition, they use different types of search algorithms to improve its performance in a given set, or they use problems such as stagnation in local optimization and loss of diversity in problem solving.

Furthermore, there are techniques combining the two techniques to reduce each other's weaknesses. Recently, the classification of topics has been used to study common applications between accurate and meta-historical approaches. To clarify each of these propositions, the analysis of distinct types of combinations may be accomplished according to the classifications introduced. These specific categories can be divided into design and implementation topics. The following are options for problems in different departments:

- Low-level, where an assumed function of a meta-heuristic is changed with another meta-heuristic, or high-level, in which various meta-heuristics are examined.
- Specific, which only settle a low range of problems with plenty higher rates and lower cost, or general aim.
- Respectively, in which the procedures operate in an integrated procedure, where each method executes at the same time from the rest.

For further reading you can visit:

Koza [6], Simon [7], Alatas [8], Kirkpatrick [9], Webster and Bernhard [10], Erol and Eksin [11], Rashedi et al. [12], Kaveh and Talatahari [13], Formato [14], Hatamlou [15] Kaveh and Khayatazad [16], Du and Zhuang [17], Moghaddam [18], Shah-Hosseini [19], Gao et al. [20], Tavakkoli-Moghaddam et al. [21], Drezner [22], Jaszkiewicz and Kominek [23] Lee et al. [24], Alba et al. [25], Chiu et al. [26].

The proposed procedure is to improve the parts where the scan of another one performs worse. Next sections provide more members about how the proposal works and its characteristics are described. The aim of this study is to achieve the good performance of these techniques, and find the accurate parameters for its application in basic optimization problems.

In this article, a novel technique is presented. The proposal is based on development and hybrid of WA and GA to solve location and routing problems. There are several reasons causing to develop this study:

- A stringent study of the parameters used in the algorithm has been conducted. Size of population and specific parameters of each part of the technique have been investigated to obtain the best arrangement possible.
- A number of different measurement values considered have been developed.
- This proposal is compared with new techniques with high efficiency and accuracy among other methods.
- This technique to achieve acceptable performance than that for statistical tests has been applied in order to demonstrate the significance of the results obtained by the presented technique.
- The new proposal aims to find synergies between the good exploration and hybrid of WA and GA, respectively.

The rest of the discussion is as follows. In Section 3, after a brief introduction to WA, the proposed method for hybrid this algorithm with GA is reviewed. 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.

3. MODEL FORMULATLON

3.1 Whale algorithm

The important thing about Reino Whales is their special hunting methods. These instinctive behaviors are referred to as their feeding method [27]. Nevertheless, Goldbogen et al. [28] examined this behavior using tag sensors. It should be noted that feeding a pure bubble is a special treatment that can be seen only in whales. In this research, it prevents the feeding of the mathematical spiral bubble to optimize the modeling.

The WA method assumes that the optimal solution for the current nominee is the target hunt, or near optimal. After determining the best answer, other factors increase their position in the top search engine. The following equations show this behavior:

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

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

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, T^* must be updated in repetitions. The vectors $\vec{N}$ and $\vec{M}$ are represented as follows:

$$\vec{N} = 2\vec{b} \cdot \vec{s} - \vec{b} \tag{3}$$

$$\vec{M} = 2 \cdot \vec{s} \tag{4}$$

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 meaning can be developed to a search space with n dimensions, and search factors over-cube around the optimal solution are always moving. According to what we have already mentioned, the whales attack their intended targets. This special mathematical method is as follows:

Two methods have been investigated to model the bubble net behavior of humpback whales:

1) Shrinking encircling method: This conduct is achieved by decreasing the amount of $\vec{b}$ in Eq. (3). The fluctuation domain of $\vec{N}$ is also decreased 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:

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

where, $\vec{F} = |\vec{T^*}(k) - \vec{T}(k)|$and shows the distance of the ith 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:

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

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

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

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

Algorithm1. Algorithm WA.

Exploration model implemented in WA (T^* is the position vector of the best solution obtained so far).

Initialize the Whales collection T_i(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. (1)

else if2(|N|≥1)

Select a assumptive search factor (T_{rand})

Update the location of the current search factor by Eq. (8)

end if2

elseif1(p≥0.5)

Update the location of the current search by Eq. (1)

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^*

3.2 Hybrid algorithm

Given that the implementation of the WA method on the assumed set al.one causes the execution time in the initial iterations to increase without reaching a suitable result, Therefore, the proposed method combines two methods of WA and GA have less time to reach the desired result. After that, as we detect the confidence interval of the optimal solution space, we gently increase the probability of utilization of WA, named P_{WA}, on the population.

Since the issue of extending the WA method was considered, we extend the hybrid method called HGAWA. In this method, this method uses the following update rule:

$$P_{WA} \leftarrow \beta_{WA} P_{WA} \quad (9)$$

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

Algorithm 2. Algorithm HGAWA.

Step 1 {Initialization}

choose the population value NP, the crossover application probability P_c, the mutation application probability P_m, the WA application probability P_{WA}, a ending condition and e, as the ratio of best individuals of existing population used for producing the population for the another generation.

Step 2 {Initial population}

Make an initial population P with measure NP hypothetically and let N_{elite}=[e*NP]

Step 3 g=0.

Step 4 {Creation of new generation}

Set P_{temp}=∅.

For i=1,..., (NP-N_{elite}) **do**

1. **Selection**: choose y_1 and y_2 from P.
2. **Crossover**: Do crossover on y_1 and y_2 with probability P_c to produce y_0.
3. **Mutation**: Implement mutation on y_0 with probability P_m.
4. Add y_0 to P_{temp}.

End for.

Step 5 Perform the WA algorithm on the best case of P_{temp} (If there is more than one item, select one hypothetically) with probability P_{WA}.

Step 6 Set g=1+g.

Step 7 Update P_{WA} using rule (9).

Step 8 {Create population for another generation}

Recreate P with P_{temp} and the set N_{elite} of best individuals selected from P.

Step 9 {Ending condition}

If the ending condition is satisfied **then stop, else go to Step 4**.

4. NUMERICAL EXPERIMENTS

In this section, a specific case of location problems is examined, as you will see the proposed algorithm will be very effective. In numerical programs, the assumptions of a locational problem, such as the exact amount required and results, are not often real. In fact, we investigate the location of the facility with fuzzy values for these presumptions and provide some degree of freedom to the decision-maker that permits for uncertainty in the input data. A natural technique for describing unspecified data is the usage of fuzzy data. Hence, here we describe a private location formula, that is, the fuzzy station blocker question, which is received by changes in its exact aggregation. To test the presented algorithm, we developed some issues of medium for large- scale FLP testing. Similar to the math formulation of the location problem as an integer-programming question [16], the formulation of the Fuzzy Location Problem (FLP) is given by:

$$\begin{aligned} max = Z(x,y) &= \mathcal{R}(\sum_{i\in I}\sum_{j\in J_i^+} \chi_{ij}\mu_{ij} f(c_{ij})\tilde{d}_j - \sum_{i\in I}\tilde{f}_i y_i) \\ &= \mathcal{R}\sum_{i\in I}\sum_{j\in J_i^+} \chi_{ij}\mu_{ij} f(c_{ij})\mathcal{R}(\tilde{d}_j) - \sum_{i\in I}\mathcal{R}(\tilde{f}_i) y_i \quad (10) \\ \text{s.t.} \quad & \sum_{i\in I_j^+} \chi_{ij} \le 1, \quad \forall j \in J \end{aligned}$$

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

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

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

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

Consider that the limitations (11) guarantee that when node i∈I is selected as a terminal (y_i=1), next it can service all the nodesin $J+i$, while the limitation (12) controls the number of the needed terminals.

Always, all locations of nodes and terminals were assumed selected in[−10, 10]×[−10, 10], with a uniform distribution. Consider that $m=|I|$, $n=|J|$ For each node of the problem, $j \in J, \tilde{d}j = (a_j^L, a_j^L + t_j, \alpha_j, \beta_j), j = 1, \dots, n,$ 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 $J_i^+ (i = 1, \dots, m)$ 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 Eq. (3). 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 (15).

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 used hybrid of WA and GA, the running time of WA, as the time limit for the ending condition of other method. Therefore, the efficiency of all methods is observed using the same runtime. The characteristics of the test problems and the time required for the methods are presented in Table 1. In each problem, the numerical results for finding the best solution, ie the solution that has the least relative error, is 1, and the solution with the maximum relative error, ie the worst solution, is 0, And the rest of the numerical solutions take values from 0 to 1, depending on how much they want the best solution. In other words, if the maximal relative error obtained by all methods on problem j is shown by ej, and the relative error got for algorithm i on problem j is shown 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. To demonstrate HGAWA's competitiveness in obtaining high quality solutions, we implemented other methods in all test problems using a higher value for runtime constraints. Then, we considered a number for a given method, on test problem i as follows:

Numerical results demonstrate that MVNS, HGAVNS and HGAWA methods have found the best solution in 91.67%, 5.57% and 2.78%, but other methods cannot reach the best solution. Compared to GA, MSA and MVNS methods, only MVNS and GA achieved the best solution in 75% and 25% of cases, respectively, and NHGASA was better than HGASA, Lin and Hong, and MSA was the worst.

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

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

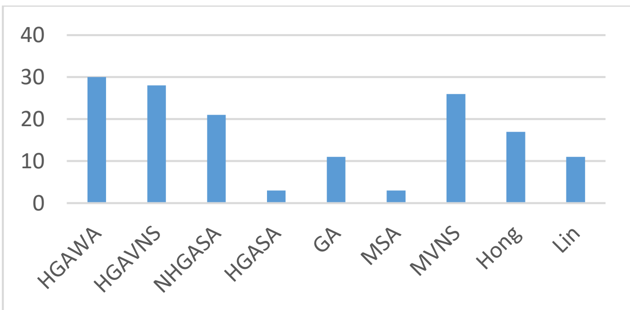

Figure 1. The scores of the methods calculated by (16)

Specifications of the test problems and the needed running times of the method are shown in Table 1 and Table 2 shows the numerical results of the calculations.

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

Table 2. Numerical results (performances)

Problem	HGAWA	NHGAVNS	HGAVNS	NHGASA	HGASA	GA	MSA	MVNS	Hong	Lin
1	1	0.7589	0.6504	0.8022	0.0682	0.8528	0	0.0217	0.7432	0.6185
2	1	0.9349	0.8702	0.8553	0	0.7484	0.0255	0.8231	0.8609	0.8376
3	1	0.9845	0.9204	0.8544	0.1847	0.571	0	0.8423	0.6752	0.6978
4	1	0.8956	0.8314	0.9426	0.1151	0.9628	0	0.6881	0.8921	0.8534
5	1	0.9846	0.9396	0.8413	0.0458	0.6229	0	0.9766	0.7904	0.6903
6	1	0.9638	0.9131	0.8234	0.0481	0.3336	0	0.9415	0.8469	0.7465
7	1	0.8593	0.739	0.8243	0.1544	0.7829	0	0.7888	0.8123	0.7424
8	1	0.9379	0.8423	0.8147	0.0683	0.5075	0	0.8623	0.7967	0.7321
9	1	0.9423	0.8629	0.8135	0.1102	0.3095	0	0.8637	0.7836	0.7023
10	0.9534	0.6342	0.4568	0.7848	0.0341	0.7871	0	0.1461	0.7601	0.7287
11	1	0.8596	0.0016	0.2608	0.2121	0	0.0129	0.2338	0.1918	0.2004
12	1	0.913	0.7518	0.7382	0.0301	0.343	0	0.7999	0.7534	0.9625
13	1	0.9678	0.8095	0.8406	0.347	0.8829	0	0.8015	0.8666	0.8209
14	1	0.9878	0.8875	0.7766	0	0.4512	0.1079	0.9191	0.6954	0.6217
15	1	0.9956	0.9432	0.7493	0.1025	0.2385	0	0.9175	0.6578	0.6625
16	1	0.9746	0.8348	0.7745	0.0229	0.6362	0	0.8332	0.651	0.6019
17	1	0.9875	0.8687	0.7407	0.0465	0.2544	0	0.8836	0.5025	0.4867
18	1	0.9689	0.9062	0.7354	0.0548	0.214	0	0.9161	0.5298	0.4806
19	1	0.6897	0	0.9334	0.7336	0.9204	0.7177	0.8166	0.9253	0.9237
20	1	0.8549	0.6902	0.717	0.0386	0.7845	0	0.6549	0.6327	0.6416
21	1	0.9456	0.7553	0.6508	0	0.2921	0.0127	0.7928	0.5907	0.3718
22	1	0.956	0.895	0.8077	0.047	0.8171	0	0.9103	0.7583	0.7509
23	1	0.9418	0.9662	0.5348	0	0.0116	0.0168	1	0.3287	0.2953
24	1	0.9534	1	0.6229	0.0404	0.1464	0	0.9909	0.5698	0.5512
25	1	0.9781	0.871	0.7323	0.0515	0.5605	0	0.8932	0.7012	0.619
26	1	0.9769	0.9035	0.6574	0.0418	0.2405	0	0.9043	0.6423	0.583
27	1	0.9985	0.9168	0.6626	0.0396	0.1508	0	0.9255	0.6217	0.6021
28	1	0.9289	0.8758	0.9464	0.5427	0.9204	0	0.8536	0.8709	0.8841
29	1	0.6988	0.5291	0.7495	0.0254	0.6807	0	0.4136	0.709	0.6823
30	1	0.6523	0.494	0.7563	0.0328	0.6597	0	0.4127	0.7713	0.7384
31	1	0.9745	0.9111	0.7315	0.0363	0.7752	0	0.985	0.7465	0.6982
32	0.9798	0.8949	0.9926	0.7203	0.0246	0.8017	0	1	0.781	0.7267
33	0.9869	0.9856	0.9089	0.747	0.0134	0.8123	0	0.9583	0.7606	0.7724
34	1	0.9745	0.9018	0.6636	0.0608	0.4948	0	0.8785	0.5763	0.6081
35	1	0.9563	0.8663	0.6061	0.0296	0.4784	0	0.8911	0.4792	0.421
36	1	0.9498	0.8629	0.6232	0.0212	0.4604	0	0.8647	0.5691	0.5258
Average	0.997781	0.912814	0.7881	0.7452	0.0864	0.5418	0.0248	0.789	0.6901	0.6276

5. CONCLUSIONS

In this paper, we present a new meta-heuristic technique with a hybrid algorithm using fuzzy values. We have executed our proposed algorithm and compared their proficiency with the recently implemented algorithms, which have been used in all collections compared to the neighborhood search method. We implemented the recommended algorithm and compared their workmanship to several other presented hybrid methods, which, in conflict, used the neighborhood search process on the population. To investigate the productiveness of the presented algorithm, we used our algorithm to station location models with fuzzy values. A fuzzy model was also a fuzzy number of nodes-related trippers, with lower boundaries and a predesignated boundary for the number of stations. We checked the algorithm on disparate randomly produced large size fuzzy station problems in which the cost ratios were assumed to be fuzzy values. The fuzzy target value in this problem was converted into a crisp one using an equation. Numerical experiments illustrated the efficiency of the proposed method on real size problems.

REFERENCES

[1] Laporte, G., Dejax, P.J. (1989). Dynamic location-routing problems. Journal of the Operational Research Society, 40(5): 471-482. https://doi.org/10.1057/jors.1989.74

[2] Salhi, S., Nagy, G. (1999). Consistency and robustness in location-routing. Studies in Locational Analysis, 13: 3-19.

[3] Ambrosino, D., Scutella, M.G. (2005). Distribution network design: New problems and related models. European Journal of Operational Research, 165(3): 610-624. https://doi.org/10.1016/j.ejor.2003.04.009

[4] Albareda-Sambola, M., Fernández, E., Nickel, S. (2012). Multiperiod location-routing with decoupled time scales. European Journal of Operational Research, 217(2): 248-258. https://doi.org/10.1016/j.ejor.2011.09.022

[5] Goldberg, D.E., Holland, J.H. (1988). Genetic algorithms and machine learning. Machine Learning, 3(2): 95-99.

[6] Koza, J.R. (1992). Genetic Programming II, Automatic Discovery of Reusable Subprograms. 1992: MIT Press, Cambridge, MA.

[7] Simon, D. (2008). Biogeography-based optimization. IEEE Transactions on Evolutionary Computation, 12(6): 702-713. https://doi.org/10.1109/TEVC.2008.919004

[8] Alatas, B. (2011). ACROA: Artificial chemical reaction optimization algorithm for global optimization. Expert Systems with Applications, 38(10): 13170-13180. https://doi.org/10.1016/j.eswa.2011.04.126

[9] Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P. (1983). Optimization by simulated annealing. Science, 220(4598): 671-680. https://doi.org/10.1126/science.220.4598.671

[10] Webster, B., Bernhard, P.J. (2003). A local search optimization algorithm based on natural principles of gravitation. https://repository.lib.fit.edu/handle/11141/117.

[11] Erol, O.K., Eksin, I. (2006). A new optimization method: big bang–big crunch. Advances in Engineering Software, 37(2): 106-111. https://doi.org/10.1016/j.advengsoft.2005.04.005

[12] Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S. (2009). GSA: A gravitational search algorithm. Information Sciences, 179(13): 2232-2248. https://doi.org/10.1016/j.ins.2009.03.004

[13] Kaveh, A., Talatahari, S. (2010). A novel heuristic optimization method: Charged system search. Acta Mechanica, 213(3): 267-289. https://doi.org/10.1007/s00707-009-0270-4

[14] Formato, R.A. (2007). Central force optimization: a new metaheuristic with applications in applied electromagnetics. Prog Electromagn Res., 77: 425-491.

[15] Hatamlou, A. (2013). Black hole: A new heuristic optimization approach for data clustering. Information Sciences, 222: 175-184. https://doi.org/10.1016/j.ins.2012.08.023

[16] Kaveh, A., Khayatazad, M. (2012). A new meta-heuristic method: Ray optimization. Computers & Structures, 112: 283-294. https://doi.org/10.1016/j.compstruc.2012.09.003

[17] Du, H., Wu, X., Zhuang, J. (2006). Small-world optimization algorithm for function optimization. In International Conference on Natural Computation, 264-273. https://doi.org/10.1007/11881223_33

[18] 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.

[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. https://doi.org/10.1504/IJCSE.2011.041221

[20] 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. https://doi.org/10.1016/j.cor.2007.01.001

[21] 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. https://doi.org/10.1016/j.cor.2007.10.011

[22] Drezner, Z. (2008). Extensive experiments with hybrid genetic algorithms for the solution of the quadratic assignment problem. Computers & Operations Research, 35(3): 717-736. https://doi.org/10.1016/j.cor.2006.05.004

[23] 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. https://doi.org/10.1016/S0377-2217(02)00830-5

[24] 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. https://doi.org/10.1016/j.asoc.2006.10.012

[25] Alba, E., Luque, G., Troya, J.M. (2004). Parallel LAN/WAN heuristics for optimization. Parallel Computing, 30(5-6): 611-628. https://doi.org/10.1016/j.parco.2003.12.007

[26] 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: An International Journal, 21(23): 3162-3172. https://doi.org/10.1002/hyp.6539

[27] 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. https://doi.org/10.2307/1379766

[28] 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. https://doi.org/10.1525/bio.2013.63.2.5