Mazandaran University of Science and Technology

Available online at www.iiec2017.com

13th International Conference on Industrial Engineering

(IIEC 2017)

Iran Institute of Industrial Engineering

Solving the Fixed Charge Transportation Problem in a Fuzzy Environment based on SA and WOA

Samira Sadeghi-Moghaddam [*a], Mostafa Hajiaghaei-Keshteli [b], Mehdi Mahmoodjanloo [c]

[a] Department of Industrial Engineering, University of Science and Technology of Mazandaran; Behshahr, Iran*
Tel:+98 11 3455 2000, Fax: +98 11 3455 2008, E-mail: ssmoghadam@gmail.com

[b,] Department of Industrial Engineering, University of Science and Technology of Mazandaran; Behshahr, Iran
Tel:+98 11 3455 2000, Fax: +98 11 3455 2008, E-mail: mostafahaji@mazust.ac.ir

[c] Department of Industrial Engineering, University of Science and Technology of Mazandaran; Behshahr, Iran
Tel:+98 11 3455 2000, Fax: +98 11 3455 2008, E-mail: mehdi.janloo@yahoo.com

Abstract

Fixed-charge transportation problem (FCTP) is a primary problem in supply chain management. To solve such an NP-Hard problem two Metaheuristics have been proposed. Since we could not formulate real world as an exact problem, therefore it is necessary to describe an approximate and a fuzzy model. In this paper both fixed costs and variable costs are considered as the fuzzy numbers. As a novelty approach, we created a procedure for converting the continuous numbers to the discrete numbers. In this paper two algorithms included SA and Whale Optimization Algorithm (WOA) are developed. Besides. This paper proposes new approaches in solution algorithms using both spanning tree based Prüfer Number and Priority based representation. Also, Taguchi method is used to guarantee the proper performance of algorithms. In addition, several various problems with different sizes are generated to assessment the capability of the algorithms and commercial software according to the real world case.

Keywords:
Fixed Charge Transportation Problem; Metaheuristic algorithm; Prüfer Number; Fuzzy sets.

1. Introduction

In the real world, the Fixed Charge Transportation Problem (FCTP) is one of the interesting problem especially in supply chain management and distribution systems. This problem and also its solution approaches are utilized in a vast bound of practical business, trade, and industrial utilizations. Simultaneously in the last decade, the problem received theoretical development.
The FCTP is a subaltern type of the fixed cost linear programming problem, introduced by Hirsch and Dantzig [1]. In a FCTP, a product is transported from supply locations to destination demand locations. Contrary to the general TP, the FCTP is more difficult to solve because of the fixed costs that result in discontinuities in the objective function and makes it indissoluble by the straight application of the transportation algorithms. (Clover *et al.*)[2].
It is shown that the problem is *NP*-Hard problem [1]. Because of the importance of problem, recently the attitude of researchers are the solution approaches. Also, Klose [3] explains a specific type of FCTP, the Single Source FCTP, is *NP*-Hard, which also proves the NP-Hardness of FCTP. Also divers heuristic methods were proposed to solve FCTP. There are several recent related works in the literature. The Problem is solved via spanning tree-based genetic algorithm by Hajiaghaei-Keshteli *et al.* [4]. They solved two examples of FTCP problem with various sizes via GA. Besides, El-Sherbiny and Alhamali [5] solved the same problem by a hybrid particle swarm algorithm with artificial immune learning in which a flexible particle is used instead of Prüfer number. Hajiaghaei-Keshteli et al. [6] developed the integrated scheduling of production and rail transportation. They used Genetic algorithm and Keshtel algorithm to address the problem. Also Hajiaghaei-Keshteli et al. [7] considered both production schedule and rail transportation allocation of orders to optimize customer service at minimum total cost. They developed a heuristic, two metaheuristics and some related procedures.
Furthermore the works on fuzzy environment, Molla-Alizadeh-Zavardehi *et al.* [8] presented a fuzzy fixed charge solid transportation problem by Metaheuristics. They solved the problem under a fuzzy environment via VNS and a hybrid algorithm of VNS and SA. Ebrahimnejad [9] proposed a method for solving fuzzy transportation problems (FTPs) in which the transportation costs and supply and demand are represented by non–negative LR flat fuzzy numbers solved using standard transportation simplex algorithms. Waiel [10] presented a fuzzy compromise programming approach to multi-objective transportation problem. Gao and Liu [11] presented two-phase fuzzy algorithms. Samanta and Roy [12] solved the multiobjective entropy transportation problem under a fuzzy environment. Omar and Samir [13] and Chanas and Kuchta [14] discussed the solution algorithm to solve the fuzzy transportation

problem. Ojha *et al.* [15] discussed a solid transportation problem with entropy in a fuzzy environment.

In this paper, to find the best solution, we attempt to use the spanning tree based SA and WOA. WOA is a new algorithm that has not been used in previous works. Also to be closer to real world, in this paper both fixed costs and variable costs are considered as the fuzzy numbers. As a new work, this paper proposes new approaches in solution algorithms using both spanning tree based Prüfer Number and Priority based representation. Our innovation is creating a procedure for converting the continuous numbers to the discrete numbers. Using Prüfer numbers, especially in designing chromosomes that do not need to check or a repairing procedure for feasibility is a good idea which mentioned in the literature. To achieve an effective mechanism, presenting a good way for encoding is necessary. Besides, twenty eight various problems with different sizes are generated to assessment the capability of the algorithms

In continuation of this article, the next section describes the model. The solution approaches are explained in Section 3. Section 4 presents the numerical experimental results. Finally, Section 5 provides conclusions.

2. Fuzzy Fixed Charge Transportation Problem (FFCTP)

In 1965, Zadeh [16] introduced the concept of fuzzy sets. Since the real world is a complicated system and we could not formulate it as an exact problem, therefore it is necessary to describe an approximate and a fuzzy model. In this paper fixed costs and variable costs are considered as the fuzzy numbers. To solve the model the fuzzy numbers should be converted to the crisp numbers by some methods which are called defuzzification methods. A fuzzy number could presents in two states, triangular and trapezoidal. In the present paper the triangular form is used. The triangular fuzzy number has a membership function which is presented with three number (*l*, *m*, *u*), the upper bound which is shown with u is the maximum number of the fuzzy number, the lower bound shown with l is the minimum number of the fuzzy number and the most probable number of a fuzzy number is shown with m.

To derive the conclusion and compute the final cost a defuzzification process is needed. Defuzzification process is a method to convert the fuzzy numbers back to crisp or classical numbers. There are several techniques to defuzzification: Mean of Maximum method, Center of Gravity method, the Height method, alpha cut method and et cetera. In the present paper the Gravity method is used [17]. For defuzzification the following formula is applied [17]:

$$CN = \frac{[(UE - LE) + (ME - LE)]}{3} + LE \qquad (1)$$

CN is a crisp number, UE is the maximum number, *LE* is the minimum number and *ME* is the most probable number.

For example suppose that we have a fuzzy number like (*3*, *5*, *7*). For converting it to a crisp number:

Fuzzy number: (*3*, *5*, *7*) we have:

$$CN = \frac{[(7-3)+(5-3)]}{3} + 3 = 5$$

Here the FCTP is considered as an effluence or distribution problem in which *m* supply points and *n* customer points exist. Every *m* suppliers can transfer materials to every *n* customers. Transferring the materials between origin and destination points have two costs. Single cost for transferring from origin *i* to destination *j* which is calculated for per unit of materials c_{ij} in addition to an opening cost for each rout named fixed cost f_{ij}. Each unit of supply for origin points $i=1, 2, \ldots, m$ is shown with a_i and each unit of demand for destination points $j=1, 2, \ldots, n$ is shown with b_j. Finding the courses that are opened and have the minimum cost of total costs contains fixed and variable costs, on the condition that the supply and demand are satisfied and the amount of transferred on those courses is the objective of problem. Standard FFCTP formulation is presented as follows:

Minimize (2)

$$Z = \sum_{i=1}^{m}\sum_{j=1}^{n}(\tilde{c}_{ij}.x_{ij} + \tilde{f}_{ij}.y_{ij}),$$

Subject to:

$$\sum_{i=1}^{m} x_{ij} \geq b_j \qquad j = 1,\ldots,n \qquad (3)$$

$$\sum_{j=1}^{n} x_{ij} \leq a_i \qquad i = 1,\ldots,m \qquad (4)$$

$$\forall i, j \qquad x_{ij} \geq 0$$

$$y_{ij} = 0 \quad if \quad x_{ij} = 0 \qquad (5)$$

$$y_{ij} = 1 \quad if \quad x_{ij} > 0 \qquad (6)$$

Where x_{ij} is the uncharted amount of transported on the course *(i,j)* that from origin *i* to destination *j*, $\tilde{c}_{ij}$ is the fuzzy Single cost for transferring from origin *i* to destination *j* which is calculated for per unit of materials. $\tilde{f}_{ij}$ is the fuzzy opening cost for each rout named fixed cost associated with course *(i,j)*. Each unit of supply for origin points $i=1, 2, \ldots, m$ is shown with a_i and each unit of demand for destination points $j=1, 2, \ldots, n$ is shown with . The transportation cost for transferring per unit from origin *i* to destination *j* is $c_{ij} \times x_{ij}$. In the present paper, since the unbalanced transportation problem can be transformed to a balanced transportation problem we suppose that the transportation problem is balanced. An unbalanced TP by adding a dummy origin or a dummy destination can be transformed to a balanced TP.

3. Proposed algorithms

3.1. SA Algorithm

With regard to the time complexity function and also a class of combinational optimization problems known as nondeterministic polynomial-time hard (NP-hard), Metaheuristic algorithms are applied to solve the FCTP. Simulated Annealing Algorithm was presented by Kirkpatrick [18]. Annealing is a process for giving more firmness to metals. The purpose of annealing is making crystalline structure spending less energy. To tackle an optimization problem via SA algorithm, it begins with a high temperature and then decreases the temperature gradually. In each temperature SA searches for best structure (solution) and then reduces the temperature by the time that no improvement happens (maybe in zero temperature). In higher temperatures there is more latitude to change in this algorithm (diversification), i.e. a bad neighbour is more acceptable.

SA algorithm steps:

Step1. Produce an initial solution and consider it as the best solution.

Step2. Fix an initial temperature ($T = T_0$)

Step3. Perform steps 4 to 7:

Step4. Generate a random solution in neighbourhood of previous solution and evaluate it.

Step5.

5.1. Accept the new solution if it is better and replace with the previous solution.

5.2. Accept the new solution if it is not better than the previous one by a probability (Boltzmann Function).

Step6. Update the best solution ever found and the temperature.

Step7. Go to step 3 if the updated temperature is greater than the threshold.

Step8. End

3.2. Whale Optimization Algorithm (WOA)

Population based algorithms have a common specification. All of them search the solution space in two phases: exploration and exploitation. In the first phase movement is random and the second phase searches the regions found by the first phase. Every Metahuristic algorithm should have a proper balance between exploration and exploitation. The Whale Optimization Algorithm (WOA) is presented by Seyedali Mirjalili *et al.* [19]. WOA is a recent method to solve the optimization problems that includes three operators to imagery the probe for prey, encircling prey, and bubble-net foraging behavior of humpback whales (Seyedali Mirjalili *et al.* [19]).

Seven different kinds of whales are: humpback, killer, right, Sei, Minke, finback, and blue (the biggest mammal in the world). This algorithm is about humpback whales and the way of hunting the victim by them. There is a special method to hunt in humpback whales called bubble-net feeding method. The humpback whales prefer hunt small fishes which are near the surface of water. To hunt the fishes the humpback whales create distinctive bubbles which are nine-shape or circle. This kind of hunting is a unique behavior that cannot be observed in others. When the humpback whales find a victim they circle it. The location of the victim is close to the optimum until the location will be update. The other searches is done to update the position to find the best location. Mentioned step is done with the following formula:

$$x_{(t+1)} = x_t - AD \quad (7)$$

t is the current location and A and D are coefficient vectors and x is the position vector. Pseudo code of WOA is shown in Figure 1.

```
Step 1: Initialize the whales population Xi(i = 1, 2, ..., n)
Step 2: Calculate the fitness of each search agent
Step 3: X*=the best search agent
Step 4: While (t < maximum number of iterations)
   for each search agent
   Update a, A, C, l, and p
      if1 (p<0.5)
        if2 (|A|< 1)
           Update the position of the current search agent by the Eq.7.
        else if2 (|A|≥ 1)
           Select a random search agent
           Update the position of the current search agent
        end if2
      elseif1 (p≥0.5)
           Update the position of the current search
      end if1
    end for
    Check if any search agent goes beyond the search space and amend it
    Calculate the fitness of each search agent
    Update X* if there is a better solution
    t=t+1
end while
```

Figure 1 - Pseudo-code of the WOA algorithm.

3.3. Continuous to discrete procedure

Our innovation in this article is creating a procedure for converting the continuous numbers to the discrete numbers. This procedure is proper for some of the algorithms such as PSO that work in continuous space and need to convert a random key solution to Prüfer Number. By using this procedure could be easily produced discrete solution in the continuous space. The following structure describes the procedure operation:

Procedure 1: Converting a random key solution to Prüfer Number
Input: X; A random key solution (a vector) with the length of m+n-2.
Output: P; Prüfer number (a vector).
Step 1: Find the rank of each cell according to its value in ascending order, and put it in the index vector.
Step 2: Find value of difference; VD:

$$VD = 0.5 * \left(max\left(X\left(index\ (1{:}\, n-1)\right)\right) + min\left(X\left(index(n{:}\, m+n-2)\right)\right)\right) \tag{8}$$

Step 3: Form the P, by the following calculations:

$$P\left(index(1{:}\, n-1)\right) = ceil\left(X\left(index(1{:}\, n-1)\right) * (1/VD) * m\right), \tag{9}$$

$$P(index(n{:}\, m+n-2)) = ceil((X(index(n{:}\, m+n-2)) - VD) * (1/(1-VD)) * n) + m. \tag{10}$$

3.4. Spanning tree-based SA and WOA algorithms

Some of the algorithms use spanning tree for finding the best rout as an important step to solve the problem. For example to decrease the cost of problems such power networks, cable connections, cubing and especially transportation problems, the mentioned networks shown as spanning tree. Two representation ways for solving various network problems are the Prüfer number and the priority based representation. Gen and Cheng [20] introduced the use of Prüfer number representation. Gen and Cheng employed the Prüfer number because is profit for representing all possible trees in a network graph (Gen and Cheng, [21]). They mentioned that the use of the Prüfer number is more proper for encoding a spanning tree, especially in some research basis like some expanded TPs (Syarif and Gen, [22]), production/distribution problem (Gen and Syarif, [23] ; Syarif, Yun, and Gen, [24]), minimum spanning problems, and so on (Gen and Cheng, [21] ; Syarif and Gen, [25]).although , the priority-based encoding had successfully been applied on transportation problem (Gen et al.)[26]. Gen et al. [27] To avoid from repair mechanisms required proposed an algorithm for two-stage transportation problem Called priority-based encoding. Although Lotfi and Tavakkoli-Moghaddam [27] used a priority based encoding for solving the FCTP via GA (pb-GA) for linear and nonlinear FCTP. They modify a priority-based decoding procedure proposed by Gen et al. [26]. They also proposed new operators for more exploration.

3.5. Initialization via Prüfer number

Creating an initial solution in algorithms are mostly a random procedure. To make an initial population, the Prüfer number generate $m+n-2$ random digits amongst [1, m + n]. In previous works before decoding the Prüfer number into the spanning tree the feasibility was checked because it is possible that the generated solution by Prüfer number cannot be adapted into the transportation network graph. Gen and Cheng [21] developed the feasibility criterion for the Prüfer number. Their technique cannot generate the feasible chromosome when the difference between the number of source nodes and the number of demand nodes is very big (Jo et al. [28]). However, Jo et al. [28], developed another feasibility criterion to check the feasibility of the chromosomes and then used a repairing procedure for infeasible chromosomes. Their repairing procedure may take long time to repair. For these reasons, there is a method to generate Prüfer number at random in which does not need a repairing procedure. (Hajiaghaei-Keshteli et al. [4])
The used feasibility criterion is as follows:

$$\sum_{i=1}^{m}(L_i+1) = \sum_{i=m+1}^{m+n}(L_i+1) \tag{11}$$

Where L_i is the appearance number of node i in Prüfer number P(T). The criterion can be showed by equation 2:

$$\sum_{i=1}^{m} L_i + m = \sum_{i=m+1}^{m+n} L_i + n \tag{12}$$

Considering the length of Prüfer number, the following equation is obtained:

$$\sum_{i=1}^{m} L_i + \sum_{i=m+1}^{m+n} L_i = m+n-2 \tag{13}$$

So we can easily show the feasibility criteria as follows:

$$\sum_{i=1}^{m} L_i = n-1 \tag{14}$$

And

$$\sum_{i=m+1}^{m+n} L_i = m-1 \tag{15}$$

A Prüfer number has $m+n-2$ digits. Considering the feasibility criteria (equations (14) and (15)), we randomly generate a string with $n-1$ digits from set O, and another with $m-1$ digits from set D. To design a feasible chromosome, the two produced strings are combined together at random.
After generating a feasible Prüfer number, the transportation network graph can be determined by using the decoding procedure that Convert Prüfer number to the transportation tree shown in Figure 2.

1	P(T) = original Prüfer number , P'(T) = the set of all the nodes that are not part of		
2	i= the lowest numbered eligible node in P'(T) , j= the leftmost digit of P(T)		
3	Are not i and j in the same set O or D?	Yes	Add the edge (i, j) to tree T
		No	Select the next digit k from P(T) that is not included in the same set with i, exchange j with k, and add the edge (i,k) to the tree T
4	Remove j (or k) from P(T) and i from P'(T). If j does not occur anywhere in the remaining part of P(T), put it intoP'(T)		
5	Assign the available amount of units to xij = min{ai,bj} to the edge (i, j) or (i,k)		
6	Update availability ai = ai–xij and bj = bj–xij (or bk = bk–xik)		
7	If no digits remain in P(T) then there are exactly two nodes, i and j, still eligible in P'(T) for consideration. Add edge (i, j) to tree T and form a tree with edges		
8	If there are no available units to assign, then stop. Otherwise, there are y plants with units and z costumers with demands yet. One of these states occurs:		
	8. 1 .If Y=1 and Z=1 , Add the edge between the plant and the customer to the tree and assign the available amount to the edge		
	8. 2. If Y>1 and Z=1 , Add the edge between the plants and the customer to the tree and assign the available amount to the edge		
	8. 3. If Y=1 and Z>1 , Add the edge between the plant and the customers to the tree and assign the available amount to the edge		
	8. 4. If and , Consider them as a new transportation model with y plants and z customers, then generate Prüfer number, and Repeat step 1 to 4		

Figure 2 - Determining the transportation network graph via Prüfer number.

3.6. Initialization via priority based

In addition to Prüfer number, Gen and Cheng [21] created another method to encoding the problem. They applied this method successfully in project control and TSP problem. In this method, a cell (gene) in a chromosome has two factors: one is locus and the other is allele. Locus is the situation of the gene in the structure of a chromosome to show a node and allele is the value that the gene takes to show the priority of the node. In priority-based encoding a gen is used to show a node and (source in transportation problem) and the value to show the priority of analogous node to creating a tree among the candidates. The steps of initialization via priority base is shown in Figure 3.

m= the number of supplier, n= the number of customer, a=the amount of supply, b=the amount of demand
Repeat the following steps until all supplies and demands will be allocated.

1 Create a (i+j) long random chromosome.
2 Choose the maximum number of chromosome.
3 Save the position of the max number and name it k.
4 If k<=m
    i=k and j=1,…,n
  And if k>m
    j=k-m and i=1,…,m
5 $uc_{ij} = c_{ij} + \dfrac{d_{ij}}{\min(a_i, b_j)}$
6 Choose the minimum of uc_{ij} and set m=i and n=j.
7 Allocate the minimum amount of supply and demand to the ith row and jth column.
8 Update the allocated row and column.
9 In the initial chromosome, set zero the gen that its supply or demand has been got zero.

Figure 3- Allocating algorithm based on priority based.

3.7. Selection mechanism

Since the objective is the minimization of total cost, better solutions are those results in lower objective function. The higher fitness value means the better chromosome, so the following function is applied to calculate each fitness value:

$$\text{Fitness Value} = \frac{1}{\text{Objective Function}} \quad (16)$$

Using the Roulette-Wheel selection mechanism, the higher fitness value a solution has, the more chance it has to be selected.

4. Experimental design

4.1. Taguchi parameter design

Recognizing a system or phenomena is required to do some experiments to discover the truth about them. Tuning the parameters and operators guarantees the proper performance of algorithms and calibration of parameters. Experiments are always cost and time consuming. When the number of factors in an experiment accumulatively increases, tuning all factors of each algorithm which is called full factorial design, may not be effective. Therefor the experiments should consume minimum level of time and cost and be effective. The method that satisfy our goal is design of experiments (DOE). There are several experimental design techniques as classic, Taguchi and *RSM*. Each of the mentioned methods have advocates especially Taguchi that is easier to analysis. Taguchi method has been successfully applied for a systematic approach for optimization (Taguchi [29] and Phadke [30]).

Factorial design for example in SA with three 3-level and one 4-level factors to solve 28 test problems which should be run two times, the total number of running the problem is 28×3^3×4^1×2 that is equal to 6048. In such experiments the precision of experiment is determined by statistical analysis (ANOVA). In sight of time and cost doing the 6048 number of experiments has no economic justification. In this situations retail factorial experiments are used that leads to reduce the number of experiments. This kind of experiments are been made simple and standard by Taguchi design.

Taguchi made and combined an especial groups of orthogonal arrays (OA) to present his experiments. For example for an experiment with four 3-level factors (3^4), the orthogonal arrays will be L9. A transformation of the repetition data to another value which is the measure of variation is developed by Taguchi. The transformation is the signal-to-noise (S/N) ratio, which explains why this type of parameter design is called a robust design [30].

Here, the term 'signal' denotes the desirable value and 'noise' denotes the undesirable value. So the S/N ratio indicates the amount of variation present in the response variable. Higher S/N ratio presents the better parameter combination. In the Taguchi method, the S/N ratio of the minimization objectives is as such Naderi et al. [31] and Naderi et al. [32].

S/N ratio = −10 log10 (objective function)^2

4.2. Data generation

To assess the performance of algorithms based on SA and WOA developed in this paper 7 different problem sizes that each problem has 4 problem type are considered. The value of problems are extracted from Sun *et al.* [33]. The range of fixed costs in each problem is differ from the others. The variable cost on the all problems has the similar limitation. Table 1 shows problem sizes, types, demand or supply and fixed costs ranges.

Table 1 - Fixed-charge transportation test problems characteristics.

Problem size	Total supply	Problem type	Range of variable costs		Range of fixed costs	
			Lower limit	Upper limit	Lower limit	Upper limit
10×10	10,000	A	3	8	50	200
10×20	15,000	B	3	8	100	400
15×15	15,000	C	3	8	200	800
10×30	15,000	D	3	8	400	1,600
50×50	50,000					
30×100	30,000					
50×100	50,000					

As mentioned before because of complication of real world, the problems are considered in fuzzy environment. With regard to the ranges of Table 1, 28 fuzzy problems are created in which the fixed and variable transportation costs are created in fuzzy environment

4.3. Experimental results

Twenty eight different test problems based on Taguchi method are solved two times to tune the parameters. The experiments for SA was based on L_{18} orthogonal arrays and for WOA was based on L9 orthogonal arrays. All the mentioned levels are tested and mean of mean and signal to noise graphs are indicated. The experiments are run two times and the mean of them is calculated. The data should be normalized for doing the statistical analysis experiments. The *RSD* is sometimes used for convenience but it can also give you an idea about how precise your data is in an experiment. One of the ways for normalizing is *RPD* (Relative Percent Deviation). The value of *RPD* for the data is calculated by the following formula:

$$RPD = \frac{|sol - Bestsol|}{|Bestsol|} * 100 \quad (17)$$

After tuning the parameters the best value of parameters and *RPD* are determined. The parameters with the maximum value in the mean *S/N* ratio plot are the best parameters. The S/N ratios of trials are averaged in each level and the value is shown in Figure 4. This figure shows best parameters in each algorithm.

For example in Prüfer -based SA, best parameters of factors A, B, C and D are obviously 1, 2, 3and 3, respectively.

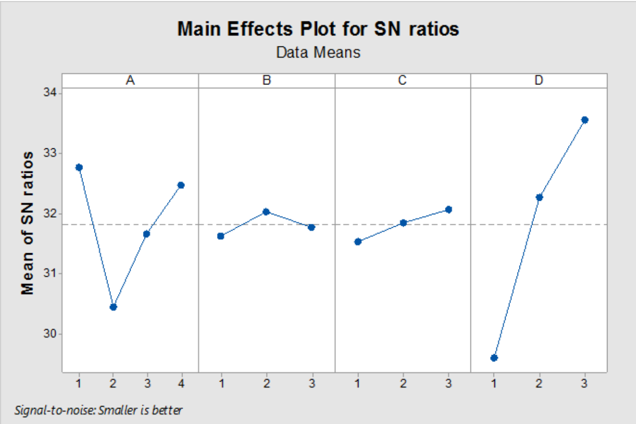

Figure 4 - Mean S/N ratio plot of Prufer-based SA.

Now for each type of encoding, Prüfer and Priority, two algorithms with the specified parameters for twenty eight different test are solved again to evaluate the best performance of algorithms based on SA and WOA.

The results of 28 problems are shown in Table 2. This table contains the results of Prüfer -based algorithms as type 1 and Priority-based algorithms as type 2. In each type two algorithms are compared and also they are compared with the near exact solution gained with GAMS to determine the gap between the exact and metaheuristic results. In Table 2, Pru indicates Prüfer and Pri indicates Priority. In this Table the best algorithm is marked with a different colour. As it could be seen Pru-WOA has the best solution amongst the other algorithm with various presentations. Also Pru-WOA has the better solution in compared with Pru-SA and Pri-WOA has the better solution in compared with Pri-SA. Generally Whale Algorithm and Prüfer presentation method has the better solution. Figure 5 shows the mean of gaps between the algorithms and the exact solution.

Table 2 – The results of 28 problems

Computational results in small, medium and large instances.
OF= Objective function; Gap=Percentage deviation from best solutions $(Z - Z^*/Z^*) \times 100$

Type	Size	Type 1 (Prüfer-based)				Type 2 (Priority-based)				Local Optimal	
		Pru-SA		Pru-WOA		Pri-SA		Pri-WOA		GAMS	
		OF	Gap	OF	Gap	OF	Gap	OF	Gap	OF	Gap
A	**10×10**	47592	3.8	47205	2.96	48470	5.72	46523	1.47	45848	0
	10×20	75908	15.75	73372	11.88	73764	12.48	73362	11.86	65582	0
	15×15	73910	13.84	73014	12.46	78530	20.95	72624	11.85	64927	0
	10×30	77394	17.53	74626	13.33	79537	20.79	75162	14.14	65848	0
	50×50	260976	63.69	244439	53.32	249856	56.72	248925	56.13	159431	0
	30×100	167207	64.10	156133	53.24	161762	58.76	158877	55.93	101891	0
	50×100	272502	69.32	260032	61.58	262482	63.10	256988	59.68	160935	0
B	**10×10**	53780	12.04	49372	2.86	51734	7.78	49122	2.34	48001	0
	10×20	79234	14.45	77397	11.80	80219	15.87	77153	11.45	69229	0
	15×15	79139	15.95	75214	10.20	81044	18.74	76599	12.23	68253	0
	10×30	84056	20.85	81185	16.72	81304	16.89	81043	16.52	69555	0
	50×50	259922	55.02	256693	53.09	267598	59.60	252648	50.68	167672	0
	30×100	178935	57.74	173670	53.10	179960	58.64	174013	53.40	113436	0
	50×100	266867	55.37	276324	60.88	282860	64.69	276827	61.17	171758	0
C	**10×10**	54590	4.38	53566	2.42	55223	5.59	54193	3.62	52301	0
	10×20	86588	13.85	82184	8.06	89337	17.47	84234	10.76	76052	0
	15×15	87732	17.32	83555	11.74	83324	11.43	83471	11.63	74778	0
	10×30	89608	16.46	88648	15.22	91688	19.17	88751	15.35	76941	0
	50×50	288368	56.72	279380	51.84	285661	55.25	280006	52.18	183997	0
	30×100	208512	56.77	204927	54.08	210116	57.98	204250	53.57	133001	0
	50×100	319879	64.38	311473	60.06	322727	65.85	307958	58.26	194593	0
D	**10×10**	63099	2.31	62176	0.81	64630	4.79	62592	1.49	61674	0
	10×20	98926	11.12	96081	7.92	103358	16.09	98315	10.43	89029	0
	15×15	98365	10.32	95468	7.07	100855	13.11	96731	8.49	89163	0
	10×30	111057	20.97	106006	15.46	107930	17.56	106629	16.14	91809	0
	50×50	332137	53.11	325193	49.91	330098	52.17	329364	51.83	216928	0
	30×100	273746	56.76	265021	51.76	272532	56.06	266817	52.79	174632	0
	50×100	395222	64.21	382919	59.10	392870	63.23	380496	58.09	240680	0
Average			33.15		29.03		33.44		29.41		0

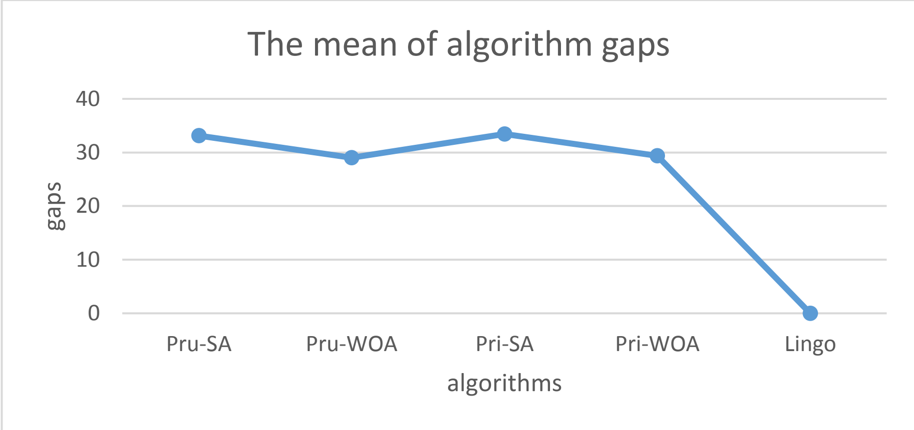

Figure 5 - The mean of gaps between the algorithms and the local optimum solution.

5. Conclusion

In the real world, the Fixed Charge Transportation Problem (FCTP) is one of the interesting problem. Because of complication of real world, the problems are considered in fuzzy environment. To derive the conclusion and compute the final cost a defuzzification process is needed. Defuzzification process is a method to convert the fuzzy numbers back to crisp or classical numbers. In the present paper the Gravity method is used. This paper created 28 problems with different sizes in fuzzy environment and two presentation method: Prüfer and Priority and solves them with two algorithms. Also these problems are solve with GAMS. In this paper we presented a method for converting continuous numbers to discrete. Besides to tune the parameters of algorithms the Taguchi method is used. This method reduces the number of experiments and the time of running the experiments. Taguchi made and combined an especial groups of orthogonal arrays (OA) to present his experiments. For example for an experiment with four 3-level factors (3^4), the orthogonal arrays will be L9. After tuning the parameters, the problems are solved and finally the results of algorithms are compared.

References

[1] Hirsch, W. M. and Dantzig, G. B, "The fixed charge problem," *Naval research logistics quarterly*, 15 (1968), 413–424.
[2] Clover. F., Klingman and D., Phillips. N.V. "Network Models in Optimization and Their Applications in Practice. *Wiley," New York*. (1992).
[3] Klose, A., "Algorithms for solving single-sink fixed-charge transportation problem," *Computers & Operations Research*, 35 (2008), 2079–2092. (2008).
[4] Hajiaghaei-Keshteli, M., Molla-Alizadeh-Zavardehi, S. and Tavakkoli-Moghaddam, R, "Addressing a nonlinear fixed-charge transportation problem using a spanning tree-based genetic algorithm," *Computers and Industrial Engineering* 59 (2010) 259–271.
[5] El-Sherbiny, M, M. and Alhamali, R, M, "A hybrid particle swarm algorithm with artificial immune learning for solving the fixed charge transportation problem," *Computers and Industrial Engineering* 64 (2013) 610–620.
[6] Hajiaghaei-Keshteli, M. and Aminnayeri, M, "Solving the integrated scheduling of production and rail transportation problem by Keshtel algorithm," *Applied Soft Computing*, 25 (2014), 184–203.
[7] Hajiaghaei-Keshteli, M., Aminnayeri, M. and Fatemi Ghomi, S.M.T, "Integrated scheduling of production and rail transportation," *Computers & Industrial Engineering*, 74 (2014), 240–256.
[8] Molla-Alizadeh-Zavardehi,S., Sadi Nezhad, S., Tavakkoli-Moghaddam,R and Yazdani, M, "Solving a fuzzy fixed charge solid transportation problem by metaheuristics," *Mathematical and Computer Modelling*, 57 (2013), 1543–1558.
[9] Ebrahimnejad, A, "New method for solving Fuzzy transportation problems with LR flat fuzzy numbers. *Information Sciences, 357* (2016), *108–124*.
[10] Waiel, F and El-Wahed, A, "A multi-objective transportation problem under fuzziness," *Fuzzy Sets and Systems*, 117 (2001), 27–33.
[11] Gao, S.P. and Liu, S.Y, "Two-phase fuzzy algorithms for multi-objective transportation problem," *The Journal of Fuzzy Mathematics*, 12 (2004), 147–155.
[12] Samanta, B. and Roy, T.K, "Multi-objective entropy transportation model with trapezoidal fuzzy number penalties, sources and destination," *Journal of Transportation Engineering*, 131 (2005), 419–428.
[13] Omar, M.S. and Samir, A.A, "A parametric study on transportation problem under fuzzy environment," *The Journal of Fuzzy Mathematics*, 11 (2013), 115–124.
[14] Chanas, S. and Kuchta, D, "A concept of the optimal solution of the transportation problem with fuzzy cost coefficients," *Fuzzy Sets and System*, 82 (1996), 299–305.

[15] Ojha, D., Mondal, S.K. and Maiti, M, "An entropy based solid transportation problem for general fuzzy costs and time with fuzzy equality," *Mathematical and Computer Modelling*, 50 (2009), 166–178.
[16] Zadeh, L.A, "Fuzzy sets," *Inf. Control*, 8 (1965), 338–353.
[17] Enric Trillas and Luka Eciolaza: Fuzzy Logic- An introductory course for engineering students, 2015th Edition *springer*.
[18] Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P, "*Optimization by simulated annealing,*" *Science*, 220 (1983), 671–679.
[19] Mirjalili, S.A. and Lewis, A, "The Whale Optimization Algorithm," *Advances in Engineering Software*, 95 (2016), 51–67.
[20] Gen, M. and Cheng, R, "Genetic algorithms and engineering design," NY: John Wiley and Sons, (2016).
[21] Gen, M. and Cheng, R. (2000), "Genetic algorithms and engineering optimization," NY: John Wiley and Sons, (2000).
[22] Syarif, A. and Gen, M, "Double spanning tree-based genetic algorithm for two stage transportation problem," *International Journal of Knowledge-based Engineering Systems*, 7 (2003b), 214–221.
[23] Gen, M., and Syarif, A, "Hybrid genetic algorithm for multi-time period production/distribution planning," *Computers and Industrial Engineering*, 48 (2005), 799–809.
[24] Syarif, A., Yun, Y. S. and Gen, M, "Study on multi-stage logistics chain network: A spanning tree based genetic algorithm," *Computers and Industrial Engineering*, 43 (2002), 299–314.
[25] Syarif, A. and Gen, M, "Solving exclusionary side constrained transportation problem by using a hybrid spanning tree-based genetic algorithm," *International Journal of Intelligent Manufacturing*, 14 (2003a), 389–399.
[26] Gen, M., Altiparmak, F. and Lin, L, "A genetic algorithm for two-stage transportation problem using priority-based encoding," *Journal of Operational Research*, 28 (2006), 337–354.
[27] Lotfi, M.M. and Tavakkoli-Moghaddam, R, "A genetic algorithm using priority-based encoding with new operators for fixed charge transportation problems," *Applied Soft Computing*, 13 (2013), 2711–2726.
[28] Jo, J.B., Li, Y. and Gen, M, "Nonlinear fixed charge transportation problem by spanning tree-based genetic algorithm," *Computers & Industrial Engineering*, 53 (2007), 290–298.
[29] Taguchi, G, "Introduction to quality engineering," *Asian Productivity Organization/UNIPUB, White Plains*, (1986).
[30] Phadke, M.S, "Quality Engineering using Robust Design," *Prentice-Hall, NJ*, (1989).
[31] Naderi, B., Zandieh, M., Ghoshe Balagh, A.K. and Roshanaei,V, "An improved simulated annealing for hybrid flowshops with sequence-dependent setup and transportation times to minimize total completion time and total tardiness," *Expert Systems with Applications* 36 (2009), 9625–9633.
[32] Naderi, B., Khalili, M. and Tavakkoli-Moghaddam, R, "A hybrid artificial immune algorithm for a realistic variant of job shops to minimize the total completion time," *Computers and Industrial Engineering* 56 (2009), 1494–1501.
[33] Sun, M., Aronson, J. E., Patrick, P. G. and Drinka, D, "A tabu search heuristic procedure for the fixed charge transportation problem," *European Journal of Operational Research* 106 (1998), 441456.