Volume 2, Number 3, 2013

International Journal of Research in Industrial Engineering

journal homepage: www.nvlscience.com/index.php/ijrie

Solving Fuzzy Step Fixed Charge Transportation Problems via Metaheuristics

F. Rajabi[1], S.E. Najafi[2], M. Hajiaghaei-Keshteli[3], S. Molla-Alizadeh-Zavardehi[4,*]

[1]Department of Industrial Engineering, Abhar Branch, Islamic Azad University, Abhar, Iran
[2]Department of Industrial Engineering, Science & Research Branch, Islamic Azad University, Tehran, Iran
[3]Department of Industrial Engineering, College of Engineering, Golestan University, Gorgan, Iran
[4]Department of Industrial Engineering, MasjedSoleyman Branch, Islamic Azad University, MasjedSoleyman, Iran

A R T I C L E I N F O

Article history :
Received:
April 18, 2013
Revised:
July 10, 2013
Accepted:
August 27, 2013

Keywords :
Transportation Problems,
Step Fixed Charge,
Simulated Annealing,
Variable Neighbourhood
Search.

A B S T R A C T

In the present paper the step fixed charge transportation problem under uncertainty, particularly when variable and fixed cost are given in fuzzy forms, is formulated. In order to solve the problem, two metaheuristic, simulated annealing algorithm (SA) and variable neighbourhood search (VNS), are developed for this NP-hard problem. Due to the significant role of parameters and operators on the algorithm's quality, an extensive calibration in both SA and VNS is carried out with the aid of a set of experimental design. Through extensive computational experiments, appropriate parameter values of the proposed algorithms were chosen. For this purpose, twenty eight problems with different configuration have been generated at random and then the effectiveness of the proposed algorithms was evaluated using the relative percentage deviation (RPD) method.

1. Introduction

The transportation problem is one of the most important areas of supply chain design that offers great potential to reduce costs. The first formulation and discussion of a classical transportation problem (TP) as an optimization problem was introduced by Hitchcock [1] with the objective of minimizing the total costs in order to transport homogeneous products from several origins to several destinations. In several decades, a vast variety of deterministic and/or nondeterministic models for TP have been developed.

A basic assumption in any transportation problem is that the cost is directly proportional to the number of units transported, while, in most real-world applications, a fixed cost for distributing of products in each route is also considered. Many practical supply chain distribution problems with fixed charge can be formulated as fixed charge transportation problems (FCTP).

Balinski [2] first formulated the FCTP and presented an approximate algorithm to solve it. Hirsch and Dantzig [3] proved that the FCTP is NP-hard. This problem is formulated as a mixed integer network programming problem and solved by some exact algorithms, such as

[*]Corresponding author
E-mail address: saber.alizadeh@gmail.com

branch and bound and cutting plane; however, these algorithms are usually inefficient and computationally expensive, especially for large-sized instances. Therefore, in the last two decades, several heuristics and metaheuristics have been presented to solve FCTPs (see, for example, heuristics [4–8]; tabu search [9]; simulated annealing [10]; genetic algorithm (GA) [11–16]; artificial immune and genetic algorithm [17]; simplex-based simulated annealing [18]; minimum cost flow-based genetic algorithm [19]).

Step fixed charge transportation problem (SFCTP) is an extended version of the FCTP and is introduced by Kowalski and Lev [20]. The SFCTP has received little attention in the transportation problem literature. To the best of our knowledge, two heuristics proposed by Kowalski and Lev [20] Altassan, et al., [21] and an artificial immune algorithm by El-Sherbiny [22] have been presented to solve SFCTPs.

In this paper, we consider the fuzzy step fixed charge transportation problem (FSFCTP). Up until now, no one has considered neither simulated annealing (SA) nor variable neighbourhood search (VNS) for any kind of FSFCTPs. So, we presented SA and VNS for solving the FSFCTP for the first time.

The rest of the paper is organized as follows. In Section 2, the FSFCTP model is described, while in Sections 3 and 4 SA and VNS are discussed. The experimental design and comparisons are presented in Section 5. Finally, the conclusion and future work are reported in Section 7.

2. Fuzzy Step Fixed Charge Transportation Problem

2.1. Preliminary

Fuzzy subsetsranking is important in the optimization problems. Various fuzzy subsets ranking approaches have been proposed in the literature (Yager [23], Dubois and Prade [24], Chen [25], Lee and Li [26]). As the problem objective function is minimizing the fuzzy transportation costs integral value, we use a simple and flexible fuzzy numbers ranking method with integral value, which is developed by Liou and Wang [27]. According to this approach, the total integral value is a convex combination of the right and left integral values through an index of optimism, $\alpha \epsilon [0, 1]$. The left integral is used to reflect the optimistic viewpoint and the right integral is used to reflect the pessimistic viewpoint of the decision maker. A convex combination of right and left integral values through an index of optimism is called the total integral value. It is used for fuzzy numbers ranking.

Triangular fuzzy number: The triangular fuzzy number is the fuzzy number with the membership function $\mu_{\tilde{A}}(x)$, a continuous mapping: $\mu_{\tilde{A}}(x): R \to [0, 1]$

$$\mu_{\tilde{A}}(x) = \begin{cases} 0 & -\infty < x < a_1 \\ \dfrac{x - a_1}{a_2 - a_1} & a_1 \le x \le a_2 \\ \dfrac{a_3 - x}{a_3 - a_2} & a_2 \le x \le a_3 \\ 0 & a_3 < x < \infty \end{cases}$$

The corresponding inverse functions of $\mu_{\tilde{A}}(x)$ can be stated by:

$$g_{\tilde{A}}(y)^L = a_1 + (a_2 - a_1)y$$

$$g_{\tilde{A}}(y)^R = a_3 - (a_3 - a_2)y = a_3 + (a_2 - a_3)y$$

where $y \in [0, 1]$. Thus, the left and right integral values are formulated as follows:

$$I(\tilde{A})^L = \int_0^1 g_{\tilde{A}}(y)^L\, dy = \frac{1}{2}(a_1 + a_2)$$

$$I(\tilde{A})^R = \int_0^1 g_{\tilde{A}}(y)^R\, dy = \frac{1}{2}(a_2 + a_3)$$

The total integral value of the TFN $\tilde{A} = (a_1, a_2, a_3)$ is

$$I_T^{\alpha}(\tilde{A}) = \alpha\, I(\tilde{A})^R + (1-\alpha) I(\tilde{A})^L = \frac{1}{2}(\alpha\, a_3 + a_2 + (1-\alpha)a_1)$$

where a degree of optimism $\alpha \in [0, 1]$ is given. When the decision degree of optimism α is 0.5, the above integral value is the same as ordinary representatives [28].

2.2. Mathematical model and descriptions

SFCTP can be stated as a transportation problem in which there are m suppliers and n customers. Each of the m suppliers can ship to any of the n customers at a shipping cost per unit fuzzy $\tilde{c}_{ij}$ plus a fuzzy fixed cost $\tilde{k}_{ij}$, assumed for opening this route. Each supplier i=1,2,…,m has S_i units of supply, and each customer j=1,2,…,n has a demand of D_j units. The objective is to determine which routes are to be opened and the size of the shipment on those routes, so that the total cost of meeting demand, given the supply constraints, is minimized. This problem can be formulated as follows:

$$\text{Min} \quad \sum_{i=1}^{m}\sum_{j=1}^{n} \tilde{c}_{ij}\, x_{ij} + \sum_{i=1}^{m}\sum_{j=1}^{n} \tilde{k}_{ij}\, y_{ij}$$

s.t

$$\sum_{i=1}^{m} x_{ij} = S_i \quad \text{i} = 1,2,\dots,\text{m},$$

$$\sum_{i=1}^{n} x_{ij} = D_j \qquad j = 1,2,\dots,n,$$

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

where

$$y_{ij} = \begin{cases} 1 & x_{ij} > 0, \\ 0 & Otherwise \end{cases}$$

The fixed cost$\tilde{k}_{ij}$ for route (*i*, *j*) is related to the transported units through its route. This consists of a fixed cost$\tilde{k}_{ij,1}$ for opening the route (*i*, *j*) and an additional cost$\tilde{k}_{ij,2}$ when the transported units exceeds a certain amount A_{ij}. Therefore,$\tilde{k}_{ij} = b_{ij,1}\tilde{k}_{ij,1} + b_{ij,2}\tilde{k}_{ij,2}$, where

$$b_{ij,1} = \begin{cases} 1 & x_{ij} > 0, \\ 0 & Otherwise \end{cases}, \quad b_{ij,2} = \begin{cases} 1 & x_{ij} > A_{ij}, \\ 0 & Otherwise \end{cases},$$

and $\tilde{k}_{ij,1}, \tilde{k}_{ij,2}, \tilde{k}_{ij}, A_{ij} \geq 0$.

Note that $\tilde{k}_{ij}$ has two steps. It could have multiple steps, depending on the problem structure. Without loss of generality, we assume that

$$\sum_{i=1}^{n} S_i = \sum_{j=1}^{m} D_j \; S_i, D_j \geq 0.$$

where x_{ij} is the unknown quantity to be transported on the route (*i*, *j*) that from plant *i* to consumer *j*, $\tilde{c}_{ij}$ is the fuzzy shipping cost per unit from plant *i* to consumer *j*. $\tilde{k}_{ij}$ is the fuzzy fixed cost associated with route (*i*, *j*). In this paper, we assume a balanced transportation problem, because the unbalanced transportation problem can be converted to a balanced transportation problem by introducing a dummy supplier or a dummy consumer. Despite its similarity to a standard TP problem, FSFCTP is significantly harder to solve because of the discontinuity in the objective function *Z* introduced by the fixed costs.

3. Metaheuristic Algorithms

3.1.Representation and Initialization

Most of the metaheuristics use a random procedure to make an initial set of solutions. The length of solution is equal to the numbers of suppliers plus customers (*m* + *n*), and then initial generation are produced with these random number in range (0, 1).

3.2.Simulated Annealing Algorithm

Simulated annealing (SA) is a widely used metaheuristic algorithm to solve combinatorial optimization problems [29–30]. In the SA an initial solution *s* is produced and then a neighboring solution s' is generated around the initial solution based on some neighborhood operator. Similar to the mutation in GA, the five neighborhood operators employed are Swap, Big Swap, Inversion, Displacement and Perturbation. As an intensification scheme, if the neighboring solution s' is better than the current solution *s*, it is accepted. As a diversification scheme, in a condition where the value of objective function of a new neighboring solution is better, SA accepts it with a probability equal to $\exp((f(s') - f(s))/T)$.

This acceptance probability is controlled by a parameter called temperature (*T*) that decreases in a regular fashion by a certain rate $\alpha \in (0,1)$ that results in a new state. When the procedure proceeds, the temperature is gradually lowered from an initial temperature (T_0) under the law of cooling schedule. Search is carried out for a fixed number of neighborhood searches in each temperature ($n_{\max}$). The algorithm is repeated until a termination condition is met. The general outline of SA is briefly described in Figure 1.

Initialization: Select an initial solution (s_0), an initial temperature (T_0), Number of neighborhood search in each temperature $n_{\max}$, and termination

---

Set $T \leftarrow T_0$ and $s \leftarrow s_0$;
Repeat
    Repeat
        Randomly select $s' \in N(s)$;
        Calculate $\Delta_{s,s'} = f(s') - f(s)$
        if $\Delta_{s,s'} \le 0$ then $s \leftarrow s'$;
        else generate random R uniformly in the range (0, 1);
        if $R \le \exp(-\Delta_{s,s'} / T_n)$ then $s \leftarrow s'$;
    Until iteration_countre = *nt*
        Decrease of the temperature *T*;
until the stopping criterion is met

Figure 1. Steps of the SA algorithm

3.3. Variable Neighbourhood Search Algorithm

Variable neighbourhood search (VNS) is a local search based metaheuristic proposed by Mladenovic [31], which involves systematic changes of neighborhoods during search process to avoid the local optimums when finding a better solutions.
Figure 2 shows the concept of VNS. VNS begins with one initial solution *x*. Then, a new solution *x′* is generated using the shaking phase $N^{s}_{k=1}$. Next, the solution is passed $n_{\max}$ times to a local search phase for finding local optima *x″*, from *x′*. The neighborhood structure used in the shake procedure is similar to the ones in the local search. In the shaking phase, it is carried out more times to make a shake. The local optima *x″* is compared with the initial solution *x*. If *x″* is better than *x*, then the search process starts with the first neighborhood structure $N^{s}_{k=1}$. Otherwise, the next neighborhood structure N^{s}_{k+1} will be used. This process repeated until $k = k_{\max}$ (i.e., $k_{\max}$ is set to be 5). Then the iteration number is incremented by one and the next iteration begins.

4. Experimental Design

4.1. Instances

Molla-Alizadeh-Zavardehi et al. [13] generated random test problems to verify the efficiency of their solution approach. We extend their plan to fuzzy and step costs in this paper. To cover various problem configurations, several levels of influencing inputs are considered. We considered all small-sized, medium-sized and large-sized problem instances, which was presented by the number of suppliers and customers. Seven different problem sizes, 10 × 10, 10 × 20, 15 × 15, 10 × 30, 50 × 50, 30 × 100 and 50 × 200 are considered for experimental design, which show different levels of difficulty for proposed solution approaches. After determining the size of test problems in a given instance, considering the important effect of the step fixed costs to the solution for each size, four problem types (A–D) are generated. For a given problem size, problem types differ from each other by the range of step fixed costs, which increases upon progressing from problem type A through problem type D. The problem sizes, types, step fixed costs ranges and their detail are shown in Table 1.

Select the set of neighbourhoodstructures N_k^s, for $k = 1,\ldots,k_{max}$, that will be used in the shaking step, and the set of neighbourhood structures N_l^{ls}, for $l = 1,\ldots,l_{max}$, that will be used in the local search step.
Number of iteration in each local search n_{max}.
Find the initial solution x.
Repeat (external loop)
    Set $k \leftarrow 1$ and $l \leftarrow 1$;
    Repeat (internal loop)
        Shaking:
        Generate random point $x' \in N_k^s(x)$;
        Local search:
        Get solution, x';
        Set $n \leftarrow 1$;
$x'' \leftarrow x'$;
        While $n \le n_{max}$ do
$x_{new} \in N_l^{ls}(x'')$;
            if $f(x_{new}) \le f(x'')$
$x'' \leftarrow x_{new}$, $n \leftarrow n + 1$;
            end
endWhile
        if $f(x'') \le f(x)$
$x \leftarrow x''$, $k \leftarrow 1$ and $l \leftarrow 1$; else $k \leftarrow k + 1$ and $l \leftarrow l + 1$;
        end
    until $k > k_{max}$ ($k_{max} = 3$)
until the stopping criterion is met

Figure 2. Steps of the VNS algorithm.

Table 1. Instances characteristics.

Problem size	Total Demand	Problem type	A_{ij}	Range of variable costs			Range of first and second fixed costs		
				a^l	a^l-b^l	α and β	a^l	a^l-b^l	α and β
10×10	10,000	A	400	U(3, 7)	U(0, 1)	U(0.25, 1)	U(50, 200)	U(0, 25)	U(5, 25)
10×20	15,000	B	400	U(3, 7)	U(0, 1)	U(0.25, 1)	U(100, 400)	U(0, 50)	U(10, 50)
15×15	15,000	C	400	U(3, 7)	U(0, 1)	U(0.25, 1)	U(200, 800)	U(0, 100)	U(20, 100)
10×30	15,000	D	400	U(3, 7)	U(0, 1)	U(0.25, 1)	U(400, 1,600)	U(0, 200)	U(40, 200)
50×50	50,000								
30×100	30,000								
50×200	50,000								

4.2. Parameter Setting

The efficiency of the metaheuristic algorithms is sensitive to the correct setting of parameter operators which affect the search process and its solution quality. Twenty-eight instances, with different sizes and specifications, are generated randomly and solved to compare the proposed algorithms.

The instances are implemented using MATLAB on a PC with dual core Duo 2 2.8 GHz and 4 GB of RAM. All algorithms ran five times and Due to having different objective functions scale in each instance their relative percentage deviation (RPD) is used. The RPD is obtained by the following formula:

$$\mathrm{RPD} = \frac{\mathrm{Alg}_{\mathrm{sol}} - \mathrm{Min}_{\mathrm{sol}}}{\mathrm{Min}_{\mathrm{sol}}} \times 100$$

where$\mathrm{Alg}_{\mathrm{sol}}$ is value of algorithm and $\mathrm{Min}_{\mathrm{sol}}$ is the best value between the algorithms. Levels of the factors are illustrated in Table 2. Using the average of RPD measures of Levels, the parameters and operators that have minimum RPD average are selected as the best ones.

Table 2. Factors and their levels.

Factors	SA symbols	VNS symbols	SA Levels	VNS Levels
Initial temperature (T_0)	A	-	A(1)- 1500	-
			A(2)- 1600	-
			A(3)- 1700	-
$n_{\max}$	B	A	B(1)-400	A (1)- 250
			B(2)-450	A (2)- 300
			B(3)-500	A (3)- 350
α	C	-	C(1)-0.91	-
			C(2)-0.92	-
			C(3)-0.93	-

The Average RPD ratios of levels are plotted against each parameter in Figs. 3 and 4. Therefore, the parameters of proposed SA were set as follows: T_0 = 1600, $n_{\max}$ = 450, α = 0.92. Also, the only factor of VNS (i. e. $n_{\max}$) was set as 300.

4.3. Experimental Results

We set searching time to be identical for both algorithms which is equal to 1.5 × (n + m) milliseconds. Hence, this criterion is affected by both n and m. The number of suppliers and customers, the more rise of searching time increases. We generated 20 instances for each twenty eight problem type, summing to 28 × 20 = 560 instances which are different from the ones used for parameter setting to avoid bias in the results. Considering twenty instances for each of the twenty eight problem type, or eighty instances for each of the seven problem sizes, for both algorithms, the instances have been run five times. Hence, using the RPD we deal with 400 data for each algorithm in each problem size.

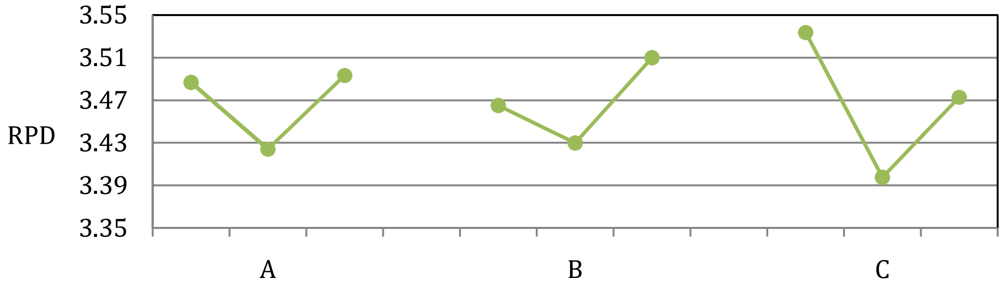

Figure 3. Average RPD for SA parameters.

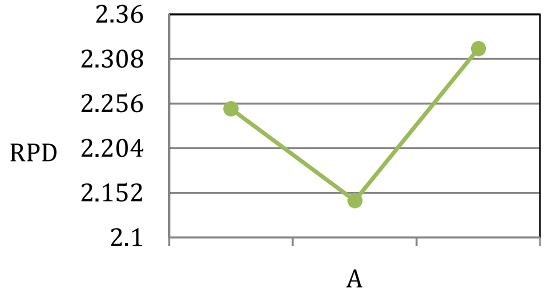

Figure 4. Average RPD for VNS parameter.

Since, we are to evaluate the robustness of the proposed algorithms in different problem sizes; the effects of the problem sizes on the performance of both SA and VNS are analyzed and compared. So, the averages of RPDs for each algorithm in each seven problem size are calculated. The interaction between the efficiency of them and the size of problems is showed in Figure 5. As can be seen from the result figure, not only is the overall performance of VNS better than SA, but it is more robust.

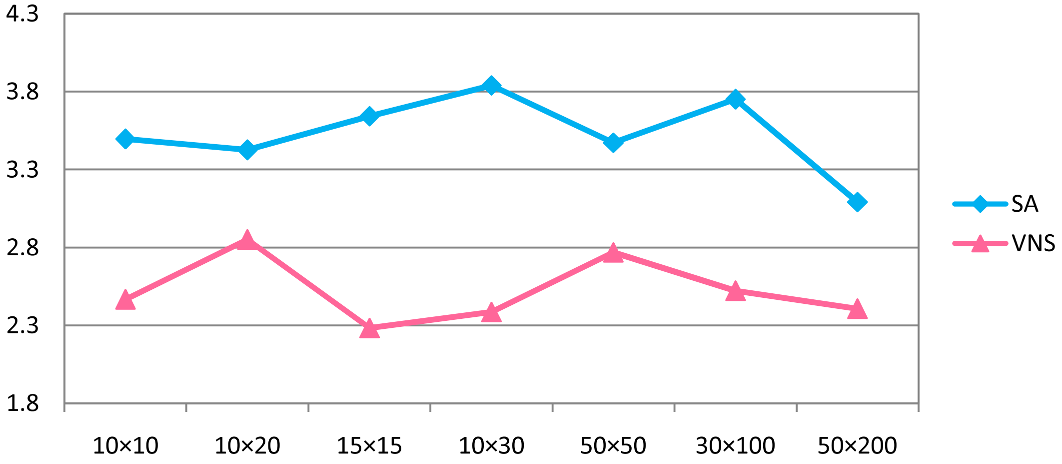

Figure 5. Interaction between proposed algorithms and problem size

5. Conclusion and Future Works

In this paper, we have proposed a SA and a VNS to solve Fuzzy Step Fixed Charge Transportation Problem. In order to evaluate the efficiency of proposed algorithm for solving the problem, a new plan is extended based on previous test problems to generate random instances. We carried out the experimental design to calibrate the parameters of these algorithms. The comprehensive set of computational experiments for instances with different configuration and difficulty show that the VNS provides good average results for test problems, all sizes, and outperforms the SA algorithm. As a direction for future research, it would be interesting to apply other fuzzy number such as generalized fuzzy number or Type-2 fuzzy for this problem. Besides, adapting multi-objective metaheuristics would be a good area for future research.

Acknowledgment

This study was partially supported by Islamic Azad University, Abhar Branch. The authors are grateful for this financial support.

References

[1] Hitchcock, F.L. (1941), The distribution of a product from several sources to numerous localities, *Journal of Mathematical Physics*, Vol. 20, pp. 224-230.
[2] Balinski, M.L. (1961), Fixed cost transportation problems, *Naval Research Logistic Quarterly*, Vol. 8, pp. 41-54
[3] Hirsch, W.M. and Dantzig, G.B. (1968), The fixed charge problem, *Naval Research Logistics Quarterly*, Vol. 15, pp. 413-424.
[4] Adlakha, V. and Kowalski, K. (2003), A simple heuristic for solving small fixed-charge transportation problems, *Omega*, Vol. 31, pp. 205-211.
[5] Eksioglu, S.D., Edwin Romeijn, H. and Pardalos, P.M. (2006), Cross-facility management of production and transportation planning problem, *Computers & Operations Research*, Vol. 33, pp. 3231-3251.
[6] Yang, L. and Liu, L. (2007), Fuzzy fixed charge solid transportation problem and algorithm, *Applied Soft Computing*, Vol. 7, pp. 879-889.
[7] Adlakha, V., Kowalski, K.,Vemuganti, R.R. and Lev, B. (2007), More-for-less algorithm for fixed-charge transportation problems, *Omega*, Vol. 35, pp. 116-127.
[8] Saez Aguado, J. (2009), Fixed charge transportation problems: a new heuristic approach based on Lagrangian relaxation and the solving of core problems, *Annals of Operations Research*, Vol. 172, pp. 45-69.
[9] Sun, M., Aronson, J.E., Mckeown, P.G. and Drinka, D. (1998), A tabu search heuristic procedure for the fixed charge transportation problem, *European Journal of Operational Research*, Vol. 106, pp. 441-456.
[10] Balaji, N. and Jawahar, N.(2010), A simulated annealing algorithm for a two-stage fixed charge distribution problem of a supply chain, *International Journal of Operational Research*, Vol. 7, pp. 192-215.
[11] Gen, M., Li, Y. and Ida, K. (2000), Spanning tree-based genetic algorithm for bi-criteria fixed charge transportation problem, *Journal of Japan Society for Fuzzy Theory and Systems*, Vol. 12, pp. 295-303.

[12] Ojha, A., Das, B., Mondal, S. and Maiti, M.(2010), A solid transportation problem for an item with fixed charge, vehicle cost and price discounted varying charge using genetic algorithm, *Applied Soft Computing*, Vol. 10, pp. 100-110.

[13] Hajiaghaei-Keshteli, M., Molla-Alizadeh-Zavardehi, S. and Tavakkoli-Moghaddam, R. (2010), Addressing a nonlinear fixed charge transportation problem using a spanning tree based genetic algorithm, *Computers & Industrial Engineering*, Vol. 59, pp. 259-271.

[14] Jawahara, N. and Balaji, N. (2012), A genetic algorithm based heuristic to the multi period fixed charge distribution problem, *Applied Soft Computing*, Vol. 12, pp. 682-699.

[15] Antony Arokia Durai Raj, K. and Rajendran, C. (2012), A genetic algorithm for solving the fixed-charge transportation model: two-stage problem, *Computers & Operations Research*, Vol. 39, pp. 2016-2032.

[16] Lotfi, M.M. and Tavakkoli-Moghaddam, R. (2013), A genetic algorithm using priority-based encoding with new operators for fixed charge transportation problems, *Applied Soft Computing*, Vol. 13, pp. 2711-2726.

[17] Molla-Alizadeh-Zavardehi, S., Hajiaghaei-Keshteli, M. and Tavakkoli-Moghaddam, R. (2011), Solving a capacitated fixed-charge transportation problem by artificial immune and genetic algorithms with a Prüfer number representation, *Expert Systems with Applications*, Vol. 38, pp. 10462-10474.

[18] Yaghini, M., Momeni, M. and Sarmadi, M. (2012), A simplex-based simulated annealing algorithm for node-arc capacitated multi commodity network design, *Applied Soft Computing*, Vol. 12, pp. 2997-3003.

[19] Xie, F. and Jia, R. (2012), Nonlinear fixed charge transportation problem by minimum cost flow-based genetic algorithm, *Computers & Industrial Engineering*, Vol. 63, pp. 763-778.

[20] Kowalski, K. and Lev, B. (2008), On step fixed-charge transportation problem, *OMEGA: International Journal Management Science*, Vol. 36, pp. 913-917.

[21] Altassan, K.M., El-Sherbiny, M.M. and Sasidhar, B. (2013), Near Optimal Solution for the Step Fixed Charge Transportation Problem, *Applied Mathematics & Information Sciences*, Vol. 7, pp. 661-669.

[22] El-Sherbiny, M.M. (2012), Alternate mutation based artificial immune algorithm for step fixed charge transportation problem, *Egyptian Informatics Journal*, Vol. 13, pp. 123-134.

[23] Yager, R.R. (1981), A procedure for ordering fuzzy subsets of the unit interval, *Information Sciences*, Vol. 24, pp. 143-161.

[24] Dubois, D. and Prade, H. (1983), Ranking fuzzy numbers in the setting of possibility theory, *Information Sciences*, Vol. 30, pp. 183-224.

[25] Chen, S.H. (1985), Ranking fuzzy numbers with maximizing set and minimizing set, *Fuzzy Sets and Systems*, Vol. 17, pp. 113-129.

[26] Lee, E.S. and Li, R.J. (1988), Comparison of fuzzy numbers based on the probability measure of fuzzy events, *Computers and Mathematics with Applications*, Vol. 15, pp. 887-896.

[27] Liou, T. and Wang, M.J. (1992), Ranking Fuzzy Numbers with Integral Value, *Fuzzy Sets and* Systems, Vol. 50, pp. 247-255.

[28] Kaufmann, A. and Gupta, M.M. (1988), Fuzzy Mathematical Models in Engineering and Management Science, *Van Nostrand Reinhold*.

[29] Kirkpatrick, S., GelattJr, C.D. andVecchi, M.P. (1983), Optimization by simulated annealing, *Science*, Vol. 220, pp. 671-680.
[30] Cerny, V. (1985), Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm, *Journal of Optimization Theory and Applications*, Vol. 45, pp. 41-51.
[31] Mladenovic, N. and Hansen, P. (1997),Variable neighborhood search, *Computers & Operations Research*, Vol. 24, pp. 1097-1100.