Solving Fuzzy Step Fixed Charge Transportation Problems via Metaheuristics

F. Rajabi¹, S.E. Najafi², M. Hajiaghaei-Keshteli³, S. Molla-Alizadeh-Zavardehi⁴,*

¹Department of Industrial Engineering, Abhar Branch, Islamic Azad University, Abhar, Iran
²Department of Industrial Engineering, Science & Research Branch, Islamic Azad University, Tehran, Iran
³Department of Industrial Engineering, College of Engineering, Golestan University, Gorgan, Iran
⁴Department of Industrial Engineering, Masjed-Soleyma Branch, Islamic Azad University, Masjed-Soleyman, Iran

Abstract

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...
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 subsets ranking 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, $\alpha \in [0, 1]$. 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: $\mu_{\tilde{A}}(x): \mathbb{R} \rightarrow [0, 1]$

$$\mu_{\tilde{A}}(x) = \begin{cases} 0 & -\infty < x < a_1 \\ \frac{x-a_1}{a_2-a_1} & a_1 \leq x < a_2 \\ \frac{a_2-x}{a_3-a_2} & a_2 \leq x < a_3 \\ \frac{a_3-x}{a_3-a_2} & a_3 < x \infty \end{cases}$$
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 $\alpha$ 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 $i=1,2,\ldots,m$ has $S_i$ units of supply, and each customer $j=1,2,\ldots,n$ 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:

$$\text{Min} \quad \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} + \sum_{i=1}^{m} \sum_{j=1}^{n} k_{ij} y_{ij}$$

s.t.

$$\sum_{j=1}^{n} x_{ij} = S_i \quad i = 1, 2, \ldots, m,$$

$$\sum_{i=1}^{m} x_{ij} = D_j \quad j = 1, 2, \ldots, n,$$

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

where

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

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

<table>
<thead>
<tr>
<th>Initialization: Select an initial solution ((s_0)), an initial temperature ((T_0)), Number of neighborhood search in each temperature (n_{max}), and termination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set ( T \leftarrow T_0 ) and ( s \leftarrow s_0 );</td>
</tr>
<tr>
<td>Repeat</td>
</tr>
<tr>
<td>\hspace{1em} Repeat</td>
</tr>
<tr>
<td>\hspace{2em} Randomly select ( s' \in N(s) );</td>
</tr>
<tr>
<td>\hspace{2em} Calculate ( \Delta_{s,s'} = f(s') - f(s) )</td>
</tr>
<tr>
<td>\hspace{2em} if ( \Delta_{s,s'} \leq 0 ) then ( s \leftarrow s' );</td>
</tr>
<tr>
<td>\hspace{2em} else generate random ( R ) uniformly in the range ((0, 1));</td>
</tr>
<tr>
<td>\hspace{2em} if ( R \leq \exp(-\Delta_{s,s'}/T) ) then ( s \leftarrow s' );</td>
</tr>
<tr>
<td>Until iteration_count ( = nt )</td>
</tr>
<tr>
<td>Decrease of the temperature ( T );</td>
</tr>
<tr>
<td>until the stopping criterion is met</td>
</tr>
</tbody>
</table>

**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 \( N_{s-1} \). 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 \( N_{s-1} \). Otherwise, the next neighborhood structure \( N_{s+1} \) 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, \( 10 \times 10 \), \( 10 \times 20 \), \( 15 \times 15 \), \( 10 \times 30 \), \( 50 \times 50 \), \( 30 \times 100 \) and \( 50 \times 200 \) are considered for
Solving Fuzzy Step Fixed Charge Transportation Problems via Metaheuristics

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 neighbourhood structures \( N_i^k \), for \( k = 1, \ldots, k_{\text{max}} \), that will be used in the shaking step, and the set of neighbourhood structures \( N_i^l \), for \( l = 1, \ldots, l_{\text{max}} \), that will be used in the local search step.

Number of iteration in each local search \( n_{\text{max}} \).

Find the initial solution \( x \).

Repeat (external loop)
  Set \( k \leftarrow 1 \) and \( l \leftarrow 1 \);
  Repeat (internal loop)
    Shaking:
      Generate random point \( x' \in N_i^k(x) \);
      Local search:
        Get solution, \( x' \);
        Set \( n \leftarrow 1 \);
    \( x'' \leftarrow x' \);
    While \( n \leq n_{\text{max}} \) do
      \( x_{\text{new}} \in N_i^l(x'') \);
      if \( f(x_{\text{new}}) \leq f(x'') \)
      \( x'' \leftarrow x_{\text{new}} \), \( n \leftarrow n + 1 \);
    endWhile
    if \( f(x'') \leq f(x) \)
      \( x \leftarrow x'' \), \( k \leftarrow 1 \) and \( l \leftarrow 1 \); else \( k \leftarrow k + 1 \) and \( l \leftarrow l + 1 \);
    end
  endforeach
until \( k > k_{\text{max}} \) (\( k_{\text{max}} = 3 \))

until the stopping criterion is met

Figure 2. Steps of the VNS algorithm.

Table 1. Instances characteristics.

<table>
<thead>
<tr>
<th>Problem size</th>
<th>Total Demand</th>
<th>Problem type</th>
<th>( A_{ij} )</th>
<th>Range of variable costs</th>
<th>Range of first and second fixed costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>10x10</td>
<td>10,000</td>
<td>A</td>
<td>400</td>
<td>U(3, 7) U(0, 1) U(0.25, 1)</td>
<td>U(50, 200) U(0, 25) U(5, 25)</td>
</tr>
<tr>
<td>10x20</td>
<td>15,000</td>
<td>B</td>
<td>400</td>
<td>U(3, 7) U(0, 1) U(0.25, 1)</td>
<td>U(100, 400) U(0, 50) U(10, 50)</td>
</tr>
<tr>
<td>15x15</td>
<td>15,000</td>
<td>C</td>
<td>400</td>
<td>U(3, 7) U(0, 1) U(0.25, 1)</td>
<td>U(200, 800) U(0, 100) U(20, 100)</td>
</tr>
<tr>
<td>10x30</td>
<td>15,000</td>
<td>D</td>
<td>400</td>
<td>U(3, 7) U(0, 1) U(0.25, 1)</td>
<td>U(400, 1,600) U(0, 200) U(40, 200)</td>
</tr>
<tr>
<td>50x50</td>
<td>50,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30x100</td>
<td>30,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50x200</td>
<td>50,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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:

\[
\text{RPD} = \left( \frac{\text{Alg}_{sol} - \text{Min}_{sol}}{\text{Min}_{sol}} \right) \times 100
\]

where \(\text{Alg}_{sol}\) is value of algorithm and \(\text{Min}_{sol}\) 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>
<thead>
<tr>
<th>Factors</th>
<th>SA symbols</th>
<th>VNS symbols</th>
<th>SA Levels</th>
<th>VNS Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial temperature ((T_0))</td>
<td>A</td>
<td>-</td>
<td>A(1)- 1500</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A(2)- 1600</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A(3)- 1700</td>
<td>-</td>
</tr>
<tr>
<td>(n_{max})</td>
<td>B</td>
<td>A</td>
<td>B(1)-400</td>
<td>A (1)- 250</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>B(2)-450</td>
<td>A (2)- 300</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>B(3)-500</td>
<td>A (3)- 350</td>
</tr>
<tr>
<td>(\alpha)</td>
<td>C</td>
<td>-</td>
<td>C(1)-0.91</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>C(2)-0.92</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>C(3)-0.93</td>
<td>-</td>
</tr>
</tbody>
</table>

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: \(T_0 = 1600\), \(n_{max} = 450\), \(\alpha = 0.92\). 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 \(1.5 \times (n + m)\) 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 \(28 \times 20 = 560\) 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.](image)

![Figure 4. Average RPD for VNS parameter.](image)

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](image)
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


---

**Solving Fuzzy Step Fixed Charge Transportation Problems via Metaheuristics**

---
