Annals of **F**uzzy **M**athematics and **I**nformatics
Volume 4, No. 1, (July 2012), pp. 9- 24
ISSN 2093–9310
http://www.afmi.or.kr

Extension of network primal simplex algorithm for solving minimum cost flow problem with fuzzy costs based on ranking functions

Ali Ebrahimnejad, Seyed Hadi Nasseri

Received 31 July 2011; Accepted 23 September 2011

Abstract. In this paper, we extend the network primal simplex algorithm for solving minimum cost flow problem which involve fuzzy numbers only in the cost coefficients using ranking function. In fact, by using linear ranking functions we present the specialization of this algorithm, known as the fuzzy network primal simplex algorithm that performs the simplex operations directly on the network without the need of a simplex tableau.

2010 AMS Classification: 90Cxx, 90C70, 90C75

Keywords: Minimum cost network problem, Network simplex algorithm, Ranking function, Trapezoidal fuzzy number.

Corresponding Author: Ali Ebrahimnejad
(a.ebrahimnejad@srbiau.ac.ir, a.ebrahimnejad@qaemshahriau.ac.ir)

1. Introduction

Fuzzy set theory has been applied to many disciplines such as control theory and management sciences, mathematical modeling and industrial applications. The concept of fuzzy mathematical programming on general level was first proposed by Tanaka et al. [12] in the framework of the fuzzy decision of Bellman and Zadeh [2]. The first formulation of fuzzy linear programming (FLP) is proposed by Zimmermann [15]. Afterwards, many authors considered various types of the FLP problems and proposed several approaches for solving these problems [4], [7, 8, 9, 10]. Some authors used the concept of comparison of fuzzy numbers for solving fuzzy linear programming problems. In effect, most convenient methods are based on the concept of comparison of fuzzy numbers by use of ranking functions [3], [7, 8, 9, 10]. Of course, ranking functions have been proposed by researchers to suit their requirements of the problem under consideration and conceivably there are no generally accepted criteria for application of ranking functions. Nevertheless, usually in such methods authors define a crisp model which is equivalent to the FLP problem and then use optimal solution of the model as the optimal solution of the FLP problem. A review of some common methods for ranking fuzzy numbers can be seen in [13]. Moreover, a review of the literature concerning fuzzy mathematical programming as well as comparison of fuzzy numbers can be seen in Klir and Yuan [5] and also Lai and Hwang [6].

In this paper we focus on solving minimum fuzzy cost network flow problem that is the most fundamental of all fuzzy network flow problems. This problem may be stated as follows: Ship the available supply through the network to satisfy demand at minimum fuzzy cost. Minimum fuzzy cost network flow problems might arise in a logistics network where people and materials are being moved between various points in the world. Clearly, the minimum fuzzy cost flow problem can be solved by fuzzy primal simplex algorithm [7, 8], [10]. But here we present the specialization of the fuzzy primal simplex algorithm to network structured fuzzy linear programming problems. This specialization, known as the network fuzzy simplex algorithm, performs the simplex operations directly on the network without the need of a simplex tableau.

This paper is organized as follows: In Section 2, we first give some necessary notations and definitions of fuzzy set theory and also some fundamental concepts of fuzzy set theory. Section 3 formulates the minimum fuzzy cost flow (MFCF) problem. Computing the basic feasible solution corresponding to a rooted spanning tree is given in Section 4. We compute the dual fuzzy variables corresponding to a rooted spanning tree in Section 5. The ptimality conditions for the minimum fuzzy cost flow problem is given in Section 6. Section 7 shows that how we can determine the exiting arc and do pivot operations on network. We extend the fuzzy network primal simplex algorithm in Section 8 and we explain it by an illustrative example. Finally, we conclude in Section 9.

2. Preliminaries

We review the fundamental notions of fuzzy set theory, initiated by Bellman and Zadeh [2].

Definition 2.1 ([8]). A convex fuzzy set $\tilde{A}$ on $\mathbb{R}$ is a fuzzy number if the following conditions hold:

(a) Its membership function is piecewise continuous.
(b) There exist three intervals $[a, b]$, $[b, c]$ and $[c, d]$ such that $\mu_{\tilde{A}}$ is increasing on $[a, b]$, equal to 1 on $[b, c]$, decreasing on $[c, d]$ and equal to 0 elsewhere.

Remark 2.2. Let $\tilde{A} = (a^L, a^U, \alpha, \beta)$ denote the trapezoidal fuzzy number, where $(a^L - \alpha, a^U + \beta)$ is the support of $\tilde{A}$ and $[a^L, a^U]$ its core. We denote the set of all trapezoidal fuzzy numbers by $F(\mathbb{R})$.

Now, we define arithmetic on trapezoidal fuzzy numbers. Let $\tilde{a} = (a^L, a^U, \alpha, \beta)$ and $\tilde{b} = (b^L, b^U, \gamma, \theta)$ be two trapezoidal fuzzy numbers. Define,

$$x > 0,\ x \in \mathbb{R};\quad x\,\tilde{a} = (xa^L, xa^U, x\alpha, x\beta)$$
$$x < 0,\ x \in \mathbb{R};\quad x\,\tilde{a} = (xa^U, xa^L, -x\beta, -x\alpha)$$
$$\tilde{a} + \tilde{b} = (a^L + b^L, a^U + b^U, \alpha + \gamma, \beta + \theta).$$

One convenient approach for solving the fuzzy linear programming problems is based on the concept of comparison of fuzzy numbers by use of ranking functions (see [7, 8], [10]). An effective approach for ordering the elements of $F(\mathbb{R})$ is to define a ranking function $R : F(\mathbb{R}) \to \mathbb{R}$ which maps each fuzzy number into the real line, where a natural order exists.

We define orders on $F(\mathbb{R})$ by:

$$\tilde{a} \succeq \tilde{b} \quad \text{if and only if} \quad R(\tilde{a}) \geqslant R(\tilde{b}) \tag{2.1}$$

$$\tilde{a} \succ \tilde{b} \quad \text{if and only if} \quad R(\tilde{a}) > R(\tilde{b}) \tag{2.2}$$

$$\tilde{a} \simeq \tilde{b} \quad \text{if and only if} \quad R(\tilde{a}) = R(\tilde{b}) \tag{2.3}$$

where $\tilde{a}$ and $\tilde{b}$ are in $F(\mathbb{R})$. Also we write $\tilde{a} \preceq \tilde{b}$ if and only if $\tilde{b} \succeq \tilde{a}$.

We restrict our attention to linear ranking functions, that is, a ranking function R such that

$$R(k\tilde{a} + \tilde{b}) = kR(\tilde{a}) + R(\tilde{b}) \tag{2.4}$$

for any $\tilde{a}$ and $\tilde{b}$ belonging to $F(\mathbb{R})$ and any $k \in \mathbb{R}$.

Remark 2.3. For any trapezoidal fuzzy number $\tilde{a}$, the relation $\tilde{a} \succeq \tilde{0}$ holds, if there exist $\varepsilon \geqslant 0$ and $\alpha \geqslant 0$ such that $\tilde{a} \succeq (-\varepsilon, \varepsilon, \alpha, \alpha)$. We realize that $R(-\varepsilon, \varepsilon, \alpha, \alpha) = 0$ (we also consider $\tilde{a} \simeq \tilde{0}$ if and only if $R(\tilde{a}) = 0$). Thus, without loss of generality, throughout the paper we let $\tilde{0} = (0, 0, 0, 0)$ as the zero trapezoidal fuzzy number.

The following lemma is now immediate.

Lemma 2.4 ([8])**.** *Let R be any linear ranking function. Then,*

(i) $\tilde{a} \succeq \tilde{b}$ *if and only if* $\tilde{a} - \tilde{b} \succeq \tilde{0}$ *if and only if* $-\tilde{b} \succeq -\tilde{a}$.
(ii) *If $\tilde{a} \succeq \tilde{b}$ and $\tilde{c} \succeq \tilde{d}$, then $\tilde{a} + \tilde{c} \succeq \tilde{b} + \tilde{d}$.*

We consider the linear ranking functions on $F(\mathbb{R})$ as:

$$R(\tilde{a}) = c_L a^L + c_U a^U + c_\alpha \alpha + c_\beta \beta, \tag{2.5}$$

where $\tilde{a} = (a^L, a^U, \alpha, \beta)$, and $c_L, c_U, c_\alpha, c_\beta$ are constants, at least one of which is nonzero. A special version of the above linear ranking function was first proposed by Yager [14] (see also [3] and [11]) as follows:

$$R(\tilde{a}) = \frac{1}{2} \int_0^1 (\inf \tilde{a}_\lambda + \sup \tilde{a}_\lambda)\, d\lambda \tag{2.6}$$

which reduces to

$$R(\tilde{a}) = \frac{a^L + a^U}{2} + \frac{1}{4}(\beta - \alpha). \tag{2.7}$$

Then, for trapezoidal fuzzy numbers $\tilde{a} = (a^L, a^U, \alpha, \beta)$ and $\tilde{b} = (b^L, b^U, \gamma, \theta)$, we have

$$\tilde{a} \succeq \tilde{b} \quad \text{if and only if} \quad a^L + a^U + \frac{1}{2}(\beta - \alpha) \geqslant b^L + b^U + \frac{1}{2}(\theta - \gamma). \tag{2.8}$$

3. Minimum cost flow problems with fuzzy costs

A definition of the minimum cost flow problem (MCFP) is given in the below.

Definition 3.1. A directed network is a directed graph whose nodes and/or arcs have associated numerical values (typically, costs, capacities, and /or supplies and demands).

Let $G = (V, E)$ be a directed network defined a set V of n nodes and a set E of m directed arcs. Each arc $(i, j) \in E$ has an associated fuzzy cost $\tilde{c}_{ij}$ that denotes the cost per unit flow on that arc. We associate with each $i \in V$ a number b_i representing its supply/demand. If $b_i > 0$, node i is a supply node; if $b_i < 0$, node i is a demand node with a demand of $-b_i$; if $b_i = 0$, node i is a transshipment node. The decision variables in the minimum fuzzy cost flow problem are arc flows and we represent the flow on an arc $(i, j) \in E$ by x_{ij}. The minimum fuzzy cost flow problem is an optimization model formulated as follows:

$$\min \quad \tilde{z} \simeq \sum_{(i,j)\in E} \tilde{c}_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{\{j:(i,j)\in E\}} x_{ij} - \sum_{\{j:(j,i)\in E\}} x_{ji} = b_i, \quad \text{for all } i \in V, \tag{3.1}$$

$$x_{ij} \geq 0, \qquad \text{for all } (i, j) \in E. \tag{3.2}$$

where $\sum_{i=1}^{n} b_i = 0$. Constraints (3.1) and (3.2) are called the mass balance constraints and nonnegative constraints, respectively. In the mass balance constraints, $\sum_{\{j:(i,j)\in E\}} x_{ij}$ represents the total flow out of node i while $\sum_{\{j:(j,i)\in E\}} x_{ji}$ indicates the total flow into node i, $\sum_{\{j:(i,j)\in E\}} x_{ij} - \sum_{\{j:(j,i)\in E\}} x_{ji}$ should be equal to b_i.

Definition 3.2. Any flow (choices of the $x_{ij}{}'s$) satisfying constraints (3.1) and (3.2) is called a feasible flow (solution).

Definition 3.3. For the minimum fuzzy cost flow problem stated in (3.1), we associate the fuzzy variable $\tilde{w}_i$ with the mass balance constraint of node i. So, the dual of minimum fuzzy cost flow problem can be stated as follows:

$$\begin{aligned} \max \quad & \tilde{y} \simeq \sum_{i\in N} \tilde{w}_i \, b_i \\ \text{s.t.} \quad & \tilde{w}_i - \tilde{w}_j \preceq \tilde{c}_{ij}, \qquad \text{for all } (i, j) \in E. \end{aligned} \tag{3.3}$$

In matrix form, we represent (MFCF) problem as follows:

$$\begin{aligned} \min \quad & \tilde{z} \simeq \tilde{c}x \\ \text{s.t.} \quad & A\,x = b \\ & x \geq 0. \end{aligned} \tag{3.4}$$

In this formulation, A is an $n \times m$ matrix, called the node-arc indicate matrix of the minimum fuzzy cost flow problem. The matrix A has one row for each node and one column for each arc of the network. Each column of A contains exactly two nonzero entries: a "+1" and a "-1". The column associated with arc (i, j) contains a "+1" in row i, a "-1" in row j and zeros elsewhere. Thus the columns of A are given by $a_{ij} = e_i - e_j$, where e_i and e_j are unit vectors in E^n, with 1's in the ith and jth positions respectively. Figure 1 presents a network with fuzzy costs.

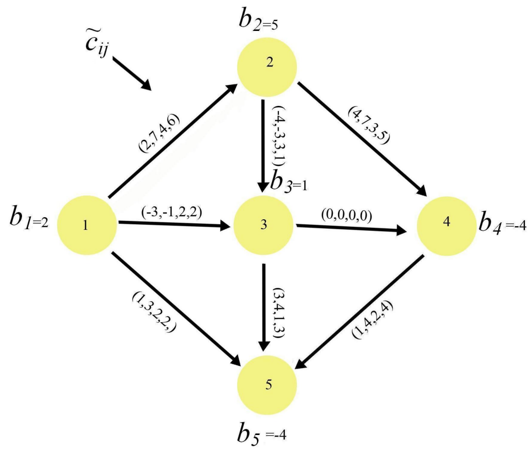


Figure 1. An example network with fuzzy costs

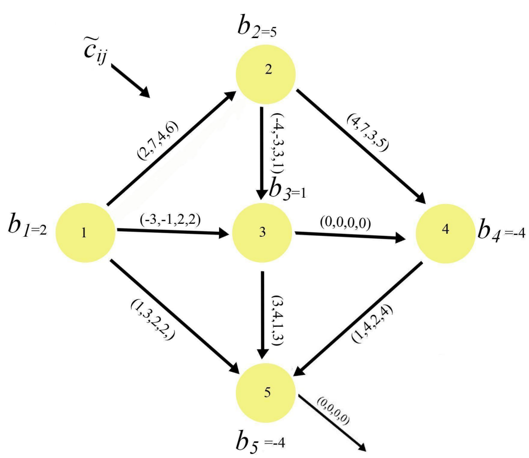


Figure 2. A rooted network

Clearly the matrix A does not have full rank since the sum of its rows is the zero vector. It can be demonstrated that the rank of A is $n - 1$ (see [1]). Therefore an artificial variable is required so that the rank of the new matrix becomes $n-1$. Introducing an artificial variable corresponding to node n leads to the constraint matrix $(A,\, e_n)$. Because, any basic solution must contain n linearly independent columns, and hence the artificial variable must appear in every basic solution. If we liberalize our definition of an arc, then the new column can be viewed as an arc beginning at node n and terminating in space (see Figure 2). This one-ended arc is called a root arc. The associated node (n) is called a root node.

Theorem 3.4 (Basic property [1]). *Consider a minimum (fuzzy) cost network flow problem defined on a connected network G with one root arc. Then B is a basic matrix for this problem if and only if it is the node-arc incidence matrix of a rooted spanning tree T of G.*

For example, for the minimum fuzzy cost flow problem in Figure 2, the rooted spanning tree T is given in Figure 3.

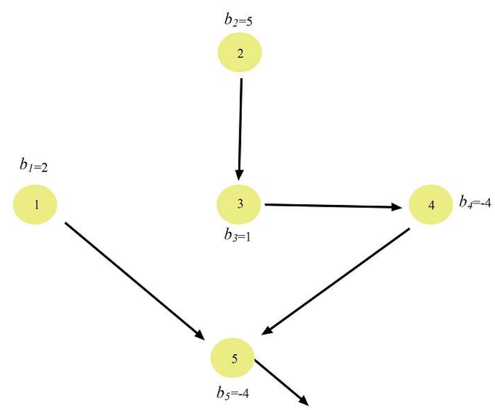


Figure 3. The rooted spanning tree

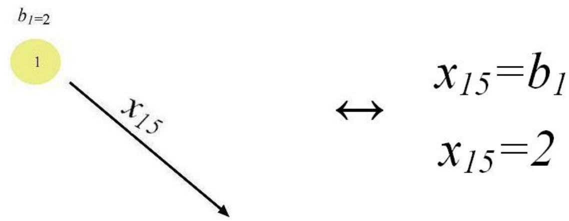
Definition 3.5. We say the basic solution corresponding to a rooted spanning tree T is feasible, if its associated solution satisfy the nonnegative constraints.

Given a rooted spanning tree T (feasible basis), the network fuzzy simplex algorithm performs the following operations:
1. Determine the associated basic feasible solution.
2. Determine the associated dual fuzzy variables.
3. Check whether it is optimal, and if not then determine an entering nonbasic arc (p, q).
4. Represent the nonbasic arc (p, q) in terms of the basic arc to perform the pivot operation while introducing the arc (p, q) into the rooted spanning tree.

We consider these fuzzy simplex operation one by one.

4. Computing the basic feasible solution

For the moment we shall postpone the difficulties associated with identifying a feasible basic and assume that a feasible basis is at hand. Since, the MFCF problem has the same feasible region as minimum (crisp) cost flow problem, therefore the calculation procedures are the same. That is, the process of obtaining the basic solution corresponding to a rooted spanning tree T, proceeds from the ends of the tree toward the root by using the mass balance constraints. For example to computing primal variables of the rooted spanning tree T given by the subgraph in Figure 3, examining node 1, we see that it is an end of the basic tree. Hence the corresponding mass balance constraints as follows:



Similarly, we can compute the rest of the basic variables. Figure 4 represents the basic feasible solution corresponding to the rooted spanning tree T given in Figure 3.

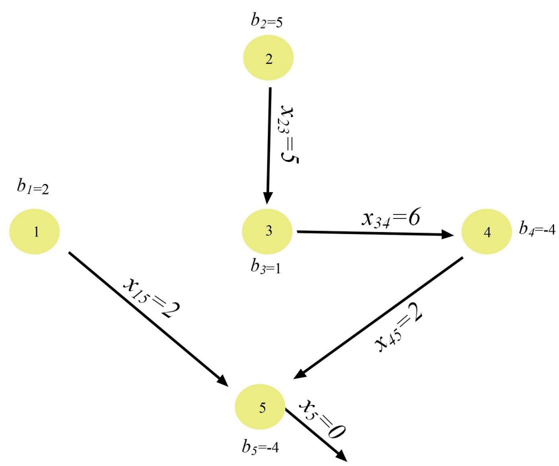


Figure 4. The basic feasible solution

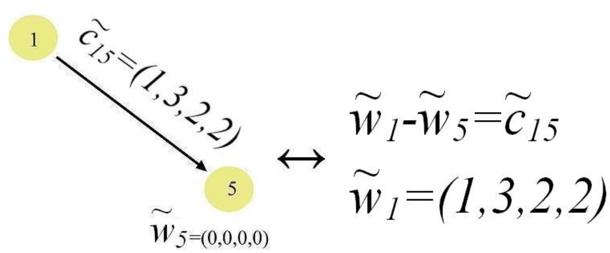
We have assume thus far that we have a starting basic feasible solution represented by a rooted spanning tree T. We now give a method for generally attaining this situation. We add a dummy node, $n+1$, to network as the root node with $b_{n+1} = 0$.Then, we add artifical arcs from each node i with $b_i \geqslant 0$ to the dummy node and from dummy node to each node i with $b_i < 0$. So, the new network arc has n new arcs, one artifical arc between each original node and the dummy node. A feasible basis for this new problem is given by that rooted spanning tree that is defined by the n artcifical arcs in addition to the root arc.

Beginning with this artificial basis, we may proceed to apply the fuzzy two-phase method, using fuzzy cost $\tilde{1} \simeq (1, 1, 0, 0)$ for each artificial arc and fuzzy cost $\tilde{0}$ for each original arc, until feasibility is achieved, if at all. If feasibility is achieved, we may drop all of the artificial arcs not in the basis, keep the basic degenerated artificial arcs and continue the optimization with initial fuzzy cost for each original arc.

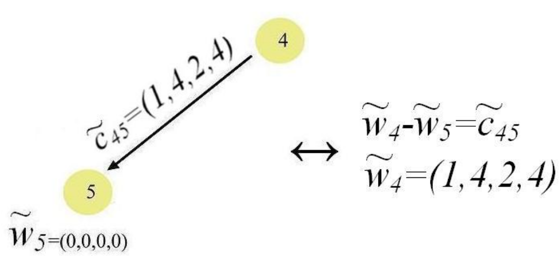
5. Computing dual fuzzy variables

While the process of computing primal variable consisted of working from the ends of the basic tree inward toward the root, the process of computing dual fuzzy variables consists of working from the root of the basic tree outward toward the ends.

We start with the dual fuzzy variable for the root node at zero fuzzy value, the proceed away from the root toward the ends of the tree using the relationship that $\tilde{w}_i - \tilde{w}_j \simeq \tilde{c}_{ij}$ along the basic arcs in the tree. In the basic tree of Figure 4, for node 1 we have



We next examine node 4. By using $\tilde{w}_4 - \tilde{w}_5 \simeq \tilde{c}_{45}$ we have



In the same fashion we can compute the rest of the dual fuzzy variables; the fuzzy values shown next to each node in Figure 5 specify these fuzzy values.

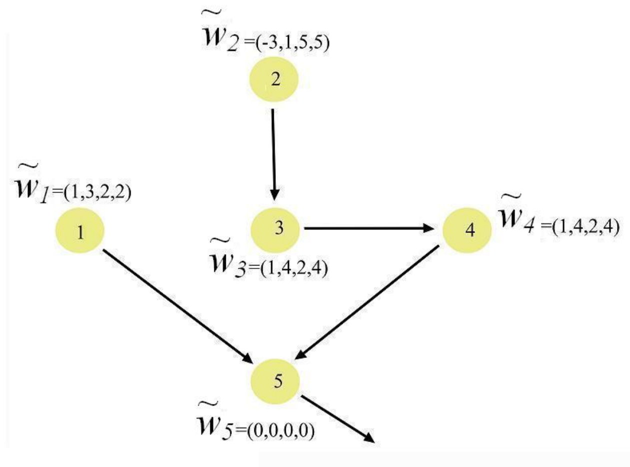


Figure 5. The dual fuzzy variables

6. Optimality testing

Given a basic tree, the fuzzy simplex method compute the dual fuzzy variables an then tests whether the basic structure satisfies the optimality conditions given in Theorem 6.1.

We define the fuzzy variable $\tilde{z}_{ij}$ for each $(i,j) \in E$ as $\tilde{z}_{ij} \simeq \tilde{c}_B B^{-1} a_{ij}$. Let $\tilde{w} \simeq \tilde{c}_B B^{-1}$ and $y_{ij} = B^{-1} a_{ij}$, hence we have $\tilde{z}_{ij} \simeq \tilde{w} a_{ij} \simeq \tilde{w}(e_i - e_j) \simeq \tilde{w}_i - \tilde{w}_j$ or $\tilde{z}_{ij} \simeq \tilde{c}_B y_{ij}$.

Theorem 6.1. (Minimum fuzzy cost network flow optimality conditions) *The feasible basic solution corresponding to rooted spanning tree T is optimum if and only if for each nonbasic arc (i,j) $\tilde{z}_{ij} \preceq \tilde{c}_{ij}$.*

Proof. Suppose that we have an optimal basic feasible solution corresponding to rooted spanning tree T, say; $\bar{x}_{ij} = \begin{cases} \bar{x}_{ij}, & (i,j) \in T \\ 0, & (i,j) \notin T \end{cases}$ whose objective fuzzy value is $\bar{z} \simeq \sum_{(i,j)\in T} \tilde{c}_{ij}\bar{x}_{ij}$. Now, let (p,q) is a nonbasic arc. The addition of arc (p,q) to the rooted spanning tree T creates the cycle C. We define the orientation of the cycle C to align with the orientation of the arc (p,q). Let C_1 and C_2 denote the sets of forward and backward arcs in C, respectively. Suppose the addition of arc (p,q) to the tree and the deleting the leaving arc (u,v) gives the following new basic feasible solution corresponding to the new rooted spanning tree T':

$$\hat{x}_{ij} = \begin{cases} \bar{x}_{ij} + \bar{x}_{uv}, & (i,j) \in C_1 \text{ and } (i,j) \neq (p,q) \\ \bar{x}_{ij} - \bar{x}_{uv}, & (i,j) \in C_2 \\ \bar{x}_{uv}, & (i,j) = (p,q) \\ 0, & \text{otherwise.} \end{cases}$$

So, the objective fuzzy value corresponding to the new basic feasible solution is equal to

$$\hat{z} \simeq \sum_{(i,j)\in T'} \tilde{c}_{ij}\hat{x}_{ij} \simeq \sum_{(i,j)\in C_1,\,(i,j)\neq(p,q)} \tilde{c}_{ij}\hat{x}_{ij} + \sum_{(i,j)\in C_2} \tilde{c}_{ij}\hat{x}_{ij} + \tilde{c}_{pq}\hat{x}_{pq}$$

$$\simeq \sum_{(i,j)\in C_1,\,(i,j)\neq(p,q)} \tilde{c}_{ij}(\bar{x}_{ij} + \bar{x}_{uv}) + \sum_{(i,j)\in C_2} \tilde{c}_{ij}(\bar{x}_{ij} - \bar{x}_{uv}) + \tilde{c}_{pq}\bar{x}_{uv}$$

$$\simeq \sum_{(i,j)\in C_1,\,(i,j)\neq(p,q)} \tilde{c}_{ij}\bar{x}_{ij} + \sum_{(i,j)\in C_2} \tilde{c}_{ij}\bar{x}_{ij} - \Big(\sum_{(i,j)\in C_2} \tilde{c}_{ij} - \sum_{(i,j)\in C_1,\,(i,j)\neq(p,q)} \tilde{c}_{ij} - \tilde{c}_{pq}\Big)\bar{x}_{uv}$$

$$\simeq \sum_{(i,j)\in C_1,\,(i,j)\neq(p,q)} \tilde{c}_{ij}\bar{x}_{ij} + \sum_{(i,j)\in C_2} \tilde{c}_{ij}\bar{x}_{ij} - \Big(\sum_{(i,j)\in C_2} \tilde{c}_{ij} - \sum_{(i,j)\in C_1} \tilde{c}_{ij}\Big)\bar{x}_{uv}.$$

Since

$$\bar{z} \simeq \sum_{(i,j)\in T} \tilde{c}_{ij}\bar{x}_{ij} \simeq \sum_{(i,j)\in C_1,\,(i,j)\neq(p,q)} \tilde{c}_{ij}\bar{x}_{ij} + \sum_{(i,j)\in C_2} \tilde{c}_{ij}\bar{x}_{ij}$$

$$\tilde{z}_{pq} \simeq \sum_{(i,j)\in C_2} \tilde{c}_{ij} - \sum_{(i,j)\in C_1} \tilde{c}_{ij}$$

and $\bar{x}_{uv} = \hat{x}_{pq}$, we have

$$\hat{z} \simeq \bar{z} - (\tilde{z}_{pq} - \tilde{c}_{pq})\hat{x}_{pq} \tag{6.1}$$

Now, from (6.1) it is obvious that if for any nonbasic arc (p, q), we have $\tilde{z}_{pq} \succ \tilde{c}_{pq}$, then we can enter (p, q) into the rooted spanning tree T and obtain $\hat{z} \prec \bar{z}$. This is contradiction to the basic feasible solution corresponding to the rooted spanning tree T is optimal. Also, if for each nonbasic arc (p, q), $\tilde{z}_{pq} \preceq \tilde{c}_{pq}$, we have $\hat{z} \succeq \bar{z}$ for any feasible solution and so the current basic feasible corresponding to the rooted spanning tree T is optimal. □

Now to compute $\tilde{z}_{ij} - \tilde{c}_{ij}$ for the nonbasic arc (i, j) we apply the definition $\tilde{z}_{ij} - \tilde{c}_{ij} \simeq \tilde{w}_i - \tilde{w}_j - \tilde{c}_{ij}$.

Thus, the $\tilde{z}_{ij} - \tilde{c}_{ij}$ can be conveniently computed on the network. In the foregoing example using the fuzzy values of dual variables obtained in previous section, we summarize the value of $\tilde{z}_{ij} - \tilde{c}_{ij}$ for each nonbasic arc (i, j) in Figure 6.

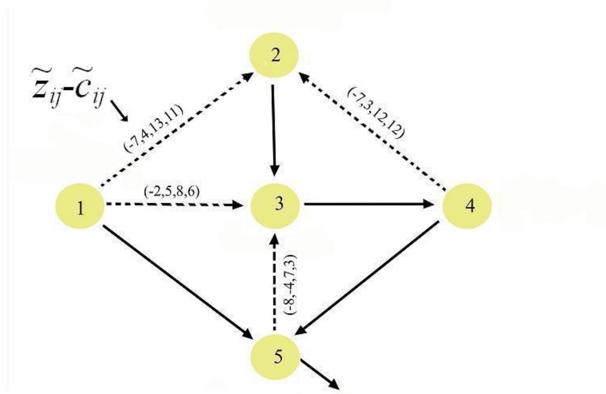


Figure 6. The value of $\tilde{z}_{ij} - \tilde{c}_{ij}$

Note also that $\tilde{z}_{ij} - \tilde{c}_{ij} \simeq \tilde{c}_B y_{ij} - \tilde{c}_{ij}$. By use of this relation, we give another method to compute $\tilde{z}_{ij} - \tilde{c}_{ij}$ for each nonbasic arc (i, j). We see, the addition of arc (i, j) to the spanning tree T creates exactly one cycle, say C. We define the orientation of the cycle C to align with the orientation of the arc (i, j). Let C_1 and C_2 denote the sets of forward and backward arcs in C, respectively. So, y_{ij} has a "$+1$" corresponding to basic arcs in C_1, a "-1" corresponding to basic arcs in C_2 and zero otherwise. Therefore, we can compute

$$\tilde{z}_{ij} - \tilde{c}_{ij} \simeq \tilde{c}_B y_{ij} - \tilde{c}_{ij} \simeq \sum_{(i,j)\in C_2} \tilde{c}_{ij} - \sum_{(i,j)\in C_1} \tilde{c}_{ij}.$$

For example $\tilde{z}_{13} - \tilde{c}_{13} \simeq \tilde{c}_{15} - \tilde{c}_{34} - \tilde{c}_{45} - \tilde{c}_{13} \simeq (-2, 5, 8, 6)$. Although this gives another method of computing $\tilde{z}_{ij} - \tilde{c}_{ij}$, it is less efficient the earlier method of computing the dual fuzzy variables $\tilde{w}$.

7. Determining the existing column and pivoting

If the spanning structure satisfies optimality conditions, it is optimal and the algorithm terminates. Otherwise, suppose

$$\mathcal{R}(\tilde{z}_{pq} - \tilde{c}_{pq}) = \max\{\mathcal{R}(\tilde{z}_{ij} - \tilde{c}_{ij}) \,:\, (i, j) \notin T\}$$

If $\mathcal{R}(\tilde{z}_{pq} - \tilde{c}_{pq}) \leq 0$, then stop; the current solution is optimal. Otherwise, we select arc (p, q) as the entering arc. The addition of this arc to the tree T creates exactly one cycle, say C. We define the direction of the cycle as the same as of the arc (p, q). We send an additional amount of flow Δ around the unique cycle created when the nonbasic arc is added to the basic tree T. Sending flow against the direction of an arc corresponds to decreasing flow on the arc. Consequently, the maximum flow Δ that can be send along this cycle is

$$\Delta = x_{uv} = \min\{x_{ij} : \text{the direction of } (i, j) \in C \text{ is against the direction of arc (p,q)}\}.$$

Thus, arc (u, v) is existing arc.

In the forgoing example, $\mathcal{R}(\tilde{z}_{13} - \tilde{c}_{13}) = 1 > 0$ and so arc $(1, 3)$ is a candidate to enter the basic tree. If we increase x_{13} by Δ, then to provide balance we must increase x_{34} by Δ, increase x_{45} by Δ, and finally decrease x_{15} by Δ. As x_{13} increases by Δ, the only basic variable to decrease is x_{15} and its new value is $x_{15} = 2 - \Delta$. Then the critical value of Δ is equal to 2, at which instant x_{15} drops to zero and $(1, 3)$ leaves the basic tree. All of the other basic variables are adjusted appropriately in value and the new basic solution is given as follows.

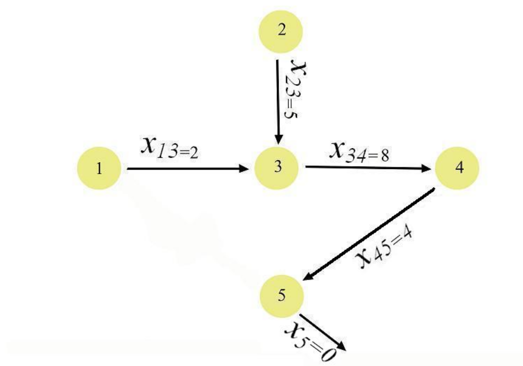


Figure 7. The new basic feasible solution

To compute the discussion of an iteration of the network fuzzy simplex method, we show how the fuzzy dual variables may be update rather than recomputed from scratch with respect to the new basic tree. Suppose that x_{pq}, enters the basic and x_{uv} is the exiting variable. The deletion of the arc (u, v) from the current basic tree partitions the set of nodes into two subtrees, one T_1, containing the root node, and the other, T_2, not containing the root node. Hence the new dual fuzzy variable for the nodes in T_1 will remain the same as before, since the chains connecting these nodes to the root node remained unchanged. For tree T_2, we consider two cases. First suppose that $q \in T_2$. Note for each $(i, j) \in T_2$, we currently have $\tilde{w}_i - \tilde{w}_j \simeq \tilde{c}_{ij}$. If we change all the $\tilde{w}_i$'s in T_2 by a fuzzy constant, we will still satisfy $\tilde{w}_i - \tilde{w}_j \simeq \tilde{\tilde{c}}_{ij}$ for all $(i, j) \in T_2$. Consequently, once we know the new variable $\tilde{w}_{q(new)}$ of the dual variable associated with node q, we can compute the new fuzzy dual value $\tilde{w}_{i(new)}$for each node i in T_2 as $\tilde{w}_i + (\tilde{w}_{q(new)} - \tilde{w}_q)$, since $\tilde{w}_{q(new)} - \tilde{w}_q)$, is the amount by which $\tilde{w}_q$ has increasing. However, denoting $\delta_{pq} \simeq \tilde{z}_{pq} - \tilde{z}_{pq} \succ \tilde{0}$, we have

$$\delta_{pq} \simeq \tilde{w}_p - \tilde{w}_q - \tilde{c}_{pq} \tag{7.1}$$

also, (p, q) is a basic arc in new tree and then $\tilde{w}_{p(new)} - \tilde{w}_{q(new)} \simeq \tilde{c}_{pq}$, and since $p \in T_1$ we have $\tilde{w}_{p(new)} \simeq \tilde{w}_q$. Hence, we have

$$\tilde{w}_p - \tilde{w}_{q(new)} \simeq \tilde{c}_{pq} \tag{7.2}$$

substituting (7.2) into (7.1) we have

$$\tilde{\delta}_{pq} \simeq (\tilde{w}_{q(new)} + \tilde{c}_{pq}) - \tilde{w}_q - \tilde{c}_{pq} \tag{7.3}$$

or, $\tilde{\delta}_{pq} \simeq \tilde{w}_{q(new)} - \tilde{w}_q$, therefore, all the dual fuzzy variables in T_2 simply increase by $\tilde{\delta}_{pq}$. On the other hand, if $p \in T_2$, then we have $\tilde{w}_{q(new)} - \tilde{w}_q \simeq \tilde{c}_{pq}$, and $\tilde{w}_{p(new)} - \tilde{w}_p \simeq -\tilde{\delta}_{pq}$. Hence in this case, all the duals of T_1 will remain the same as previously, but dual of T_2 will fall by $\tilde{\delta}_{pq}$. For example, in the foregoing pivot operation, $(p, q) = (1, 3)$ enters an $(u, v) = (1, 5)$ leaves the basic tree. Disconnecting $(1, 5)$, we find that the tree T_1 contains the nodes 2, 3, 4, and 5, but T_2 contains node 1 alone. Since $p \in T_2$, the dual of node 1 falls by $\tilde{\delta}_{13} \simeq (-2, 5, 8, 6)$ to the fuzzy value $\tilde{w}_{l(new)} \simeq (-4, 5, 8, 10)$ and the dual of the other nodes remain the same as before. Figure 8 shows the new dual fuzzy variables respect to the new basic tree given in Figure 7.

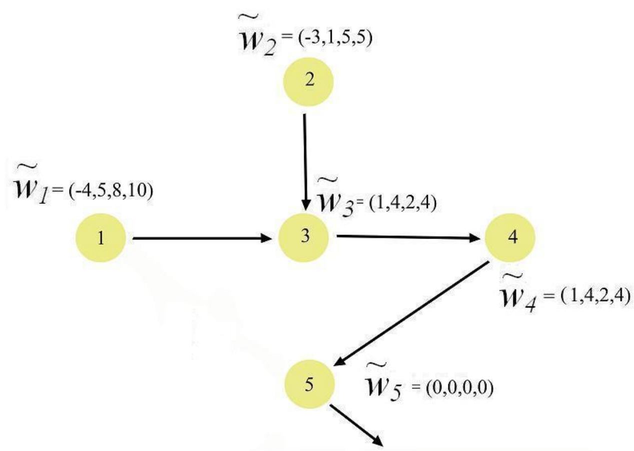


Figure 8. The updated values of the dual fuzzy variables

8. Fuzzy network primal simplex algorithm

The network fuzzy simplex algorithm maintains a feasible basic solution corresponding to rooted spanning tree and successively transforms it into an improved feasible basic solution until it becomes optimal. This algorithm may be stated as follows:

begin

Find an initial basic feasible solution represented by a rooted spanning tree. Compute the basic flows x and dual fuzzy variables $\tilde{w}$ associated with the basic tree.

While some nonbasic arc violates the optimality condition do

begin

Select an entering arc (p, q) violating its optimality condition. Add arc (p, q) to the tree and determine the leaving arc (u, v); perform a tree update and update the solution x and $\tilde{w}$.

end

end

We use the example in Figure 2, to illustrate the network fuzzy simplex algorithm. Figure 4 shows a basic feasible solution for the problem and Figure 5 shows the dual fuzzy variables corresponding to this basic feasible solution. By the values of $\tilde{Z}_{ij} - \tilde{c}_{ij}$'s given in Figure 6, arc $(1,3)$ enters and $(u,v) = (1,5)$ leaves the basic tree (Figure 7). In the new basic feasible solution shown in the Figure 7, we updated the dual fuzzy variables (Figure 8).

Now, we compute the $\tilde{z}_{ij} - \tilde{c}_{ij}$ for each nonbasic arc (i,j) in new rooted spanning tree to see whether the new spanning tree satisfy the optimality conditions. Figure 9 shows these values.

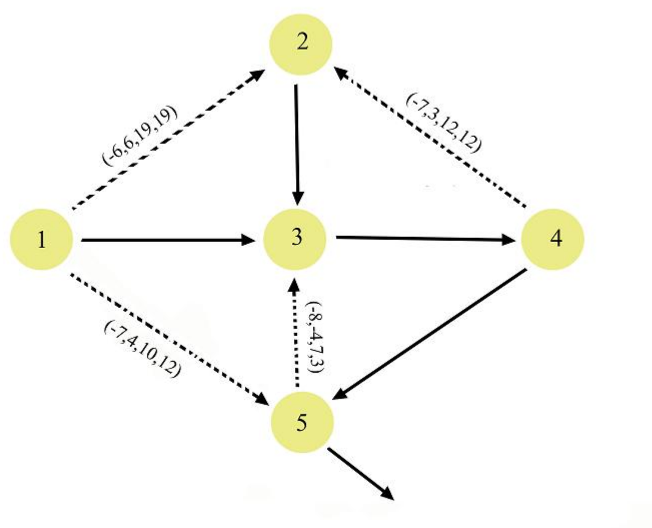


Figure 9. The updated values of $\tilde{z}_{ij} - \tilde{c}_{ij}$

In this example, since for each nonbasic arc (i,j), we have $\mathcal{R}(\tilde{z}_{ij} - \tilde{c}_{ij}) \geq 0$ then, the current solution shown in Figure 7 is optimal. The fuzzy objective function value is equal to $\tilde{z} \simeq \sum_{(i,j)\in T} \tilde{c}_{ij}\, x_{ij}$. We have

$$\tilde{z} \simeq 2\,\tilde{c}_{12} + 5\,\tilde{c}_{23} + 8\,\tilde{c}_{34} + 4\,\tilde{c}_{45} \simeq 2\,(-3,-1,2,2) + 5\,(-4,-3,3,1)$$
$$+8\,(0,0,0,0) + 4\,(1,4,2,4) \simeq (-22,-1,27,25),$$

and the its membership function is as follow

$$\mu(\tilde{z}) = \begin{cases} 0, & x < -49, \\ \frac{x+49}{27}, & -49 \leq x < -22, \\ 1, & -22 \leq x \leq -1, \\ \frac{24-x}{25}, & -1 \leq x < 24, \\ 0, & x > 24. \end{cases}$$

9. Conclusions

In this paper, we considered minimum cost flow problems which involve fuzzy numbers only in cost coefficients of objective function. Then, by use of a linear ranking function we gave a fuzzy simplex algorithm to solve the minimum fuzzy cost flow problem. We described the network fuzzy simplex algorithm as a combinatorical algorithm and used combinatorial arguments to show that the algorithm correctly solves the MFCF problem. This development has the advantage of highlights the inhernet combinatorial structure of the MFCF problem on the fuzzy simplex algorithm. The network fuzzy simplex algorithm is indeed a adaptation of the fuzzy simplex method for linear programming problems with fuzzy costs. Beacuse the MFCF problem is highly structured fuzzy linear programming problem, when we apply the fuzzy simplex method to it, the resulting computations become considerably streamlined. In fact we need not explicitly maintain the matrix the fuzzy linear program and can perform all the computations directly on the netwrok, and it is a special implementation of the fuzzy simplex method that exploits the special structure of MFCF problem.

Acknowledgements. The authors would like to thank the anonymous reviewers and honorable Editor of the journal for accepting the paper.

References

[1] M. S. Bazaraa, J. J. Jarvis and H. D. Sherali, Linear Programming and Network Flows, John Wiley, New York, Third Edition, 2005.

[2] R. E. Bellman and L. A. Zadeh, Decision making in a fuzzy environment, Management Sci. 17 (1970) 141–164.

[3] P. Fortemps and M. Roubens, Ranking and defuzzification methods based on area compensation, Fuzzy Sets and Systems 82 (1996) 319–330.

[4] K. Ganesan and P. Veeramani, Fuzzy linear programming with trapezoidal fuzzy numbers, Ann. Oper. Res. 143 (2006) 305–315.

[5] G. J. Klir and B. Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications, Prentice-Hall, PTR, New Jersey, 1995.

[6] Y. J. Lai and C. L. Hwang, Fuzzy Mathematical Programming Methods and Applications, Springer, Berlin, 1992.

[7] N. Mahdavi-Amiri and S. H. Nasseri, Duality in fuzzy number linear programming by use of a certain linear ranking function, Appl. Math. Comput. 180 (2006) 206–216.

[8] N. Mahdavi-Amiri and S. H. Nasseri, Duality results and a dual simplex method for linear programming problems with trapezoidal fuzzy variables, Fuzzy Sets and systems 158 (2007) 1961–1978.

[9] H. R. Maleki, Ranking functions and their applications to fuzzy linear programming, Far East J. Math. Sci. (FJMS) 4 (2002) 283–301.

[10] H. R. Maleki, M. Tata and M. Mashinchi, Linear programming with fuzzy variables, Fuzzy Sets and Systems 109 (2000) 21–33.

[11] M. Roubens, Inequality constraints between fuzzy numbers and their use in mathematical programming, in: R. Slowinski, J. Teghem (Eds.), Stochastic Versus Fuzzy Approaches To Multiobjective Mathematical Programming Under Uncertainty, Kluwer Academic Publishers, Dordrecht, 1991, pp. 321–330.

[12] H. Tanaka, T. Okuda and K. Asai, On fuzzy mathematical programming, J. Cybernet. 3 (1973) 37–46.

[13] X. Wang and E. Kerre, Reasonable properties for the ordering of fuzzy quantities (2 parts), Fuzzy Sets and Systems 118 (2001) 375–405.

[14] R. R. Yager, A procedure for ordering fuzzy subsets of the unit interval, Inform. Sci. 24 (1981) 143–161.

[15] H. J. Zimmermann, Fuzzy programming and linear programming with several objective functions, Fuzzy Sets and Systems 1 (1978) 45–55.

Ali Ebrahimnejad
(a.ebrahimnejad@srbiau.ac.ir, a.ebrahimnejad@qaemshahriau.ac.ir)
Department of Mathematics, Qaemshahr Branch, Islamic Azad University, Qaemshahr, Iran.

Seyed Hadi Nasseri (nasseri@umz.ac.ir)
Department of Mathematical Sciences, University of Mazandaran , Babolsar, Iran.