Novelty Journals

ISSN 2394-9651

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 4, Issue 3, pp: (1-9), Month: September - December 2017, Available at: www.noveltyjournals.com

An Implementation of Network Simplex Method (Big-M) for solving Minimum Cost Flow Problem

[1] Dr. Mohammad Hassan Hassan, [2]Farah Ghalawingy

[1]Professor, Department of Mathematics, Faculty of Science, TishreenUniversity, Lattakia, Syria
[2]Master student, Department of Mathematics, Faculty of Science, Tishreen University, Lattakia, Syria

Abstract: **The Minimum Cost Flow (MCF) Problem is to send flow from a set of supply nodes, through the arcs of a network, to a set of demand nodes, at minimum total cost, and without violating the lower and upper bounds on flows through the arcs .**

In this paper, we will illustrate an implementation of network simplex method (big-M)version for a variant of MCF.

Network simplex method (N.S.A) is one of the most popular and effective solution methods for (MCF) problem in practice. It maintains a basic feasible solution and gradually improves its objective function value by small transformations, called pivots.

We will consider four pivot rules and compare their features .

Keywords: **Minimum Cost Flow , Network flow , Network Optimization , Network Simplex (big-M), Pivot rules , Strongly feasible tree .**

1. INTRODUCTION

There is a very extensive literature on network optimization . Optimization is to find the best value of the variables that make optimal the objective function satisfying a set of constraint[3] .Network flow theory comprises a wide range of optimization models, which have countless applications in various fields . One of the most fundamental network flow problems is the minimum-cost flow (MCF) problem [4].

It aims at finding the optimal flows along the arcs of network in order to minimize the total cost subject to conservation of flow at each node. This problem directly arises in various real-world applications in the fields of transportation, logistics, telecommunication, network design, resource planning, scheduling, and many other industries[4].

The early starts of MCF formulation were in classic transportation problem . Transportation problem was first studied by a Russian mathematician, L.V. Kantorovich, in a paper entitled Mathematical Methods of Organizing and Planning Production (1939) [1], [11].

The MCF formulation is particularly broad and can be used as a template upon which a number of network problems may be modeled as shortest paths , maximum flow , assignment problem , transportation problem , transshipment problem[1], [6], [12] .

MCF Problem and the Network Simplex Method (NSA) were initially developed independently.NSA is an adaption of the bounded variable primal simplex algorithm ,in which all operations are performed on the network of the problem [1] . The LP variables correspond to the arcs of the graph and the LP bases are represented by spanning trees. The NS algorithm is devised by Dantzig, the inventor of the LP simplex method. He first solved the un capacitated transportation problem using this method and later generalized the bounded variable simplex method to directly solve the MCF problem[8].

NSA starts with initial spanning tree solution. We will use artificial initialization technique because the artificial spanning tree solution can be constructed easily and quickly [1], [2], [5] .

At each iteration, an entering arc is selected by some pricing strategy, and forms a cycle with the arcs of the tree. The leaving arc is the arc of the cycle with the least augmenting flow. The substitution of entering for leaving arc, and the reconstruction of the tree is called a *pivot*. When no non-basic arc remains eligible to enter, the optimal solution has been reached [1].

Steps of NSA will be illustrated by numerical example, and four different pivot strategies will be considered and compared briefly.

2. MINIMUM COST FLOW PROBLEM

2.1 .Minimum Cost Flow Problem Formulation:

2.1.1 Primal Case:

Let G = (N,A) be a directed network consisting of a finite set of nodes, N = {1,2…..,n}, and a set of directed arcs, A = {1,2,…..m}, linking pairs of nodes in N. We associate with every arc of (i, j) ∈A, a flow x_{ij}, a cost per unit flow c_{ij}, a lower bound on the flow l_{ij}, (we consider $l_{ij} = 0$) and a capacity u_{ij}.

To each node $i \in N$ we assign an integer number $b(i)$ representing the available supply of, or demand for flow at that node. If $b(i) > 0$ then node i is a supply node, if $b(i) < 0$ then node i is a demand node, and otherwise, where $b(i) = 0$, node i is referred to as a transshipment node. Total supply must equal total demand.

The Minimum Cost Flow (MCF) Problem is to send the required flows from the supply nodes to the demand nodes (i.e. satisfying the demand constraints (2,2)), at minimum cost. The flow bound constraints, (2,3), must be satisfied. The demand constraints are also known as flow conservation constraints.

The formulation of the problem as a Linear Programming (LP) problem is as follows:[1]

$$minimize \sum_{(i,j)\in A} c_{ij}x_{ij} \qquad (2.1)$$

Subject to

$$\sum_{j:(i,j)\in A} x_{ij} - \sum_{j:(j,i)\in A} x_{ji} = b(i) \qquad \text{for all } i \in N \qquad (2.2)$$

$$0 \le x_{ij} \le u_{ij} \qquad \text{for all } (i,j) \in A \qquad (2.3)$$

That the total net supply must equal zero can be seen by summing the flow conservation equations over all $i \in N$ resulting in

$$\sum_{j:(i,j)\in A} x_{ij} - \sum_{j:(j,i)\in A} x_{ji} = \sum_{i\in N} b(i) = 0$$

The problem is described in matrix notation as

$minimize\{ cx \,|\, Nx = b$ and $0 \le x \le u \}$ where N is a node-arc incidence matrix having a row for each node and a column for each arc.

2.1.2 Dual Case:

$$\text{maximize} \sum_{i\in N} b(i)\pi(i) \qquad (2.4)$$

Subject to $\quad \pi(i) - \pi(j) - \delta_{ij} \le c_{ij} \qquad (2.5) \quad$ for all $(i,j) \in A$

$\delta_{ij} \geq 0$ (2.6) for all $(i, j) \in A$

$\pi(j)$ is the dual variable for node j, so that $\pi(j)$is unrestricted for all j ∈ N.

$\pi(j)$ is known as the node price or node potential, for node j.[1]

2.2 Optimality Conditions:

2.2.1 Complementary Slackness Optimality Conditions

A feasible solution x^* is optimal if, and only if, for some set of node potentials π, the following is satisfied for every arc (i, j) ∈ A:

If $c_{ij}^{\pi} > 0$ then $x^*_{ij} = 0$

If $c_{ij}^{\pi} = 0$ then $0 < x^*_{ij} < u_{ij}$

If $c_{ij}^{\pi} < 0$ then $x^*_{ij} = u_{ij}$

where $c_{ij}^{\pi} = c_{ij} - \pi(i) + \pi(j)$also known as the reduced cost of arc (i, j) [2],[7] .

2.2.2 Reduced Cost Optimality Conditions:

A feasible solution x^* is an optimal solution if, and only if, some set of node potentials π satisfy the following reduced cost optimality conditions: $c_{ij}^{\pi} \geq 0$ for every $a_{ij} \in A$ [2],[7] **.**

2.2.3 Negative Cycle Optimality Conditions:

A feasible flow $x^*$is an optimum flow if and only if it admits no negative cost augmenting cycle [1].

Theorem : (Strong Duality Theorem).

For any choice of problem data, the minimum cost flow problem always has a solution x* and the dual minimum cost flow problem has a solution π satisfying the property that z(x*) = w(π) [2].

3. PRIMAL NETWORK SIMPLEX ALGORITHM (N.S.A)

3.1 Spanning Tree Structure [2],[7]:

The algorithm uses a spanning tree structure to maintain a feasible basis (the basic feasible solutions). Specific to the Network Simplex Method, define the spanning tree structure by the triplet (T, L,U), where :

T are the arcs of the spanning tree

L are the non-tree arcs whose flow is restricted to 0

U are the non-tree arcs whose flow is maximized .

A spanning tree (T, L,U) is optimal when it is feasible and satisfies:

$c_{ij}^{\pi} = 0$ for all $a_{ij} \in T$

$c_{ij}^{\pi} \geq 0$ for all $a_{ij} \in L$

$c_{ij}^{\pi} \leq 0$ for all $a_{ij} \in U$

3.2 The Algorithm [1],[2],[5],[6],[7]:

Step 0 Initialization

Find an initial basic feasible spanning tree.

We can consider node1 as root, or we add an artificial root node with zero supply$b_0 = 0$, also artificial arcs are added . Root node with artificial arcs constitute initial spanning tree for big-M method . The corresponding basic flow vector is given by:

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 4, Issue 3, pp: (1-9), Month: September - December 2017, Available at: www.noveltyjournals.com

$$x_{i0} = b_i \quad \forall i \text{ with } b_i > 0$$

$$x_{0i} = -b_i \quad \forall i \text{ with } b_i \leq 0$$

$$\boldsymbol{x_{ij}} = 0 \quad \forall (i,j) \in A$$

Step1 Check if the spanning tree is optimal

If so , stop ,otherwise , go to 2.

Step2 Select the entering arc

An arc in the non basic L or U violates optimality conditions. There are various rules to choose the entering arc such as Dantzig`s rule that chooses an arc having the minimum negative reduced cost value .

Step3 Determine the leaving arc

Adding a new arc creates cycle W, a pivot cycle . Saturate the pivot and remove the first saturated arc .

Step4 Update the spanning tree

Re compute the node potentials . Return to step2.

3.3 Strongly Feasible Trees[1] , [5]:

A degenerate pivot is one in which the flow augmentation is zero and thus the value of the objective function remains unchanged.

Such degenerate pivots only modify the spanning tree, but the flow itself remains unchanged . In this case there is no guarantee that the tree *T* will not be repeated after several degenerate iterations with no improvement in the primal cost . To avoid this possibility, thereby ensuring termination of the method, a special property is used which called **strongly feasible trees property**.

Strongly feasible trees property: A feasible tree *T* with corresponding flow vector *x* is said to be strongly feasible if every arc (i, j) of *T* with $x_{ij} = 0$ is oriented away from the root.

The artificial starting solution described above generates a strongly feasible basis, for consider each node, $i \in N$.

To implement the algorithm effectively , we need to represent the spanning trees appropriately in a computer , NSA uses elaborate storage schemes such as ATI (Augmented Threaded Index) method [1] , [2] , and XTI (Extended Threaded Index) method [9] .

3.4 Complexity:

N.S.A complexity is exponential, but there are attempts to formulate polynomial time versions. Orlin [13] presented polynomial time network primal simplex algorithm which performs $\boldsymbol{O(n^2 m \min\{\log(nC), m\log n\})}$pivots, where n is the number of nodes and m is the number of arcs. C denotes the maximum absolute arc cost**.**

3.5 Numerical Example:

We will solve a variant of M.C.F known as single source shortest path problem by N.S.A (big-M) method.

The origin of this example is in [5]. We will illustrate in details N.S.A work to achieve the optimal solution.

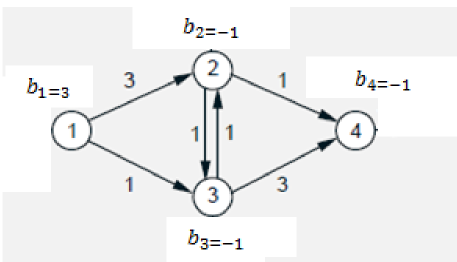


Figure 1 The problem Node1 is source, nodes 2,3,4 are sinks, the arc lengths(arc costs) are shown

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 4, Issue 3, pp: (1-9), Month: September - December 2017, Available at: www.noveltyjournals.com

The corresponding minimum cost flow problem is

The objective function : $Z_{\min} 3x_{12} + x_{13} + x_{23} + x_{24} + x_{32} + 3x_{34}$

Subject to:

$x_{12} + x_{13} = 3$

$x_{23} + x_{24} - x_{12} - x_{32} = -1$

$x_{32} + x_{34} - x_{23} - x_{13} = -1$

$-x_{24} - x_{34} = -1$

$x_{ij} \geq 0 \qquad \forall\, (i, j) \in A$

The source node 1 is selected as root , we introduce artificial arcs to connect the root with each node (i≠1) with very large cost M in order to not affect the optimal solution .Now the root node with artificial arcs constitute strongly feasible initial tree(all arcs are oriented away from the root) , (figure 2) . Every artificial arc carries unit flow , and non tree arcs with zero flow .

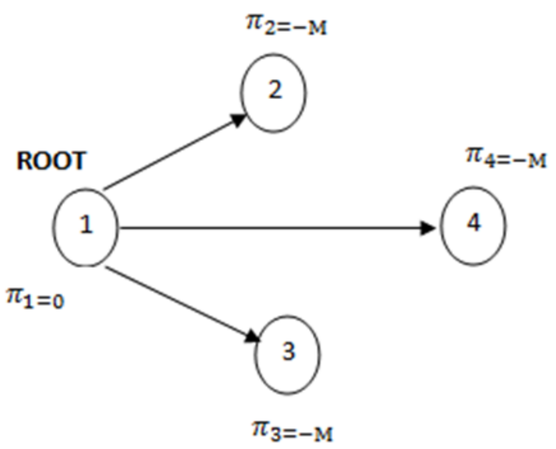


Figure 2 : initial spanning tree

We calculate the tree node prices .

For $\quad \pi_1 = 0, \quad \pi_1 - \pi_j = c_{1j} \qquad \pi_1 - \pi_2 = c_{12}, \quad 0 - \pi_2 = M\,, \quad \pi_2 = -M$

$\pi_1 - \pi_3 = c_{13}, \qquad 0 - \pi_3 = M\,, \qquad \pi_3 = -M$

$\pi_1 - \pi_4 = c_{14}, \qquad 0 - \pi_4 = M\,, \qquad \pi_4 = -M$

The corresponding price vector is [0 , -M , -M , -M] .

Reduced costs connected with non artificial arcs :

$c_{12}{}^{\pi} = c_{12} - \pi_1 + \pi_2 = 3 - M$

$c_{13}{}^{\pi} = c_{13} - \pi_1 + \pi_3 = 1 - M$

$c_{24}{}^{?} = c_{24} - \pi_2 + \pi_4 = 1 + M - M = 1$

$c_{34}{}^{\pi} = c_{34} - \pi_3 + \pi_4 = 3$

$c_{23}{}^{\pi} = c_{23} - \pi_2 + \pi_3 = 1$

$c_{32}{}^{\pi} = c_{32} - \pi_3 + \pi_2 = 1$

Novelty Journals

ISSN 2394-9651

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 4, Issue 3, pp: (1-9), Month: September - December 2017, Available at: www.noveltyjournals.com

In the first iteration we can select some arc (1,j) ∈A as an in-arc and selecting the artificial arc connecting 1 and j as out arc .

An arc (1,3) is selected as in-arc because it has the least reduced cost , (Dantzig`s rule) , and the artificial arc (1,3) as out-arc , (figure 3)

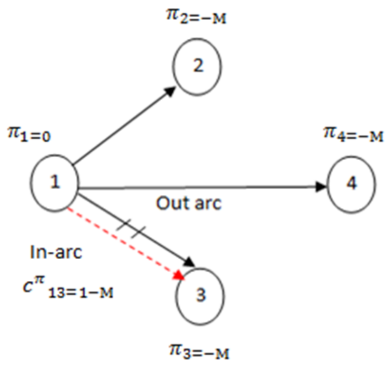


Figure 3 : first pivot

The corresponding flow $[x_{13} = 1 , x_{12} = 1 , x_{14} = 1]$.

We calculate node prices for the basis arcs

$\pi_1 - \pi_3 = c_{13}, 0 - \pi_3 = 1 , \pi_3 = -1$

$\pi_1 - \pi_2 = c_{12}, 0 - \pi_2 = M , \pi_3 = -M$

$\pi_1 - \pi_4 = c_{14}, 0 - \pi_4 = M , \pi_4 = -M$

Reduced costs for non tree arcs:

${c_{12}}^{\pi} = c_{12} - \pi_1 + \pi_2 = 3 - M$

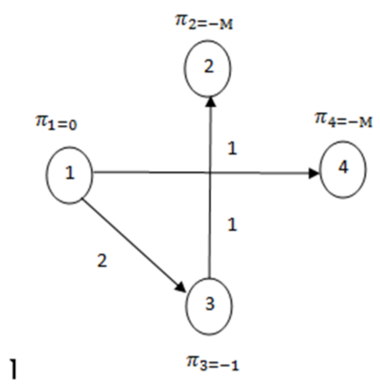
${c_{24}}^{\pi} = c_{24} - \pi_2 + \pi_4 = 1 + M - M = 1$

${c_{34}}^{\pi} = c_{34} - \pi_3 + \pi_4 = 3 + 1 - M = 4 - M$

${c_{23}}^{\pi} = c_{23} - \pi_2 + \pi_3 = 1 + M - 1 = M$

${c_{32}}^{\pi} = c_{32} - \pi_3 + \pi_2 = 2 - M$

Entering arc (3,2) forms a cycle with tree arcs . To remove it , let flow of arc (3,2) is Θ , The flow on the forward arc(1,3) is increased by Θ ,and the flow on the backward arc (1,2) is decreased by Θ , then Θ = 1 , the artificial arc (1,2) , is the leaving arc (figure 4). Figure 5 shows the second pivot.



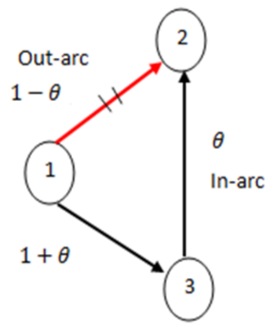


Figure 4:Selecting out- arc **Figure 5: The second pivot**

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 4, Issue 3, pp: (1-9), Month: September - December 2017, Available at: www.noveltyjournals.com

The corresponding flow $[x_{13} = 2 , x_{14} = 1 , x_{32} = 1]$

Node prices for the basis arcs

$\pi_1 - \pi_3 = c_{13}, 0 - \pi_3 = 1 , \pi_3 = -1$

$\pi_1 - \pi_4 = c_{14}, 0 - \pi_4 = M , \pi_4 = -M$

$\pi_3 - \pi_2 = c_{32}, -1 - \pi_2 = 1 , \pi_2 = -2$

Reduced costs connected with non tree arcs :

$c_{12}{}^{\pi} = c_{12} - \pi_1 + \pi_2 = 3 - 0 - 2 = 1$

$c_{24}{}^{\pi} = c_{24} - \pi_2 + \pi_4 = 1 + 2 - M = 3 - M$

$c_{34}{}^{\pi} = c_{34} - \pi_3 + \pi_4 = 3 + 1 - M = 4 - M$

$c_{23}{}^{\pi} = c_{23} - \pi_2 + \pi_3 = 1 + 2 - M = 3 - M$

In- arc is (2,4), out- arc is (1,4) . Figure 6. Shows the third pivot .

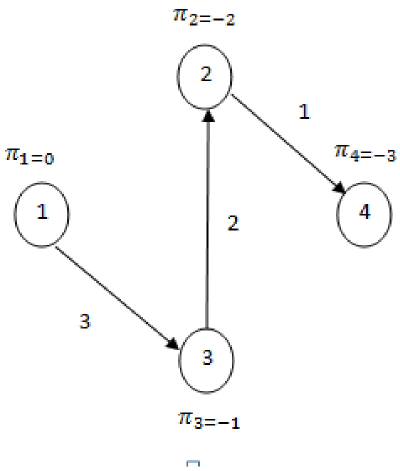


Figure 6 : Third pivot

There are no artificial arcs .

The corresponding flow $[x_{13} = 3 , x_{32} = 2 , x_{24} = 1]$.

Reduced costs for non basic arcs :

$c_{12}{}^{\pi} = c_{12} - \pi_1 + \pi_2 = 3 - 0 - 2 = 1$

$c_{34}{}^{\pi} = c_{34} - \pi_3 + \pi_4 = 3 + 1 - 3 = 1$

$c_{23}{}^{\pi} = c_{23} - \pi_2 + \pi_3 = 1 + 2 - 1 = 2$

We notice that all $c_{ij}{}^{\pi} \geq 0$ then , there is no eligible arc to enter the basis and algorithm terminates with optimal flow $[x_{13} = 3 , x_{32} = 2 , x_{24} = 1]$.

The corresponding node prices $[\pi_1 = 0 , \pi_2 = -2 , \pi_3 = -1, \pi_4 = -3]$.

Now , we apply strongly duality theorem

$$\sum_{(i,j)\in A} c_{ij}x_{ij} = c_{13}x_{13} + c_{32}x_{32} + c_{24}x_{24} = 1(3) + 1(2) + 1(1) = 6$$

$$\sum_{I\in N} b(i)\pi(i) = 3(0) - 1(-2) - 1(-1) - 1(-3) = 6$$

The pair (x^{*},π)satisfies the optimality conditions.

3.6 Pivot Rules:

One of the most conclusi**ve** issues that determines the goodness of N.S.A solution is selecting of the entering arc . We will discuss four pivot rules briefly , detailed discussion in [1] , [2] .

- **Best eligible arc pivot rule(Dantzig`s rule).**At each iteration, this method selects an eligible arc with the maximum violation to enter the basis . This forms a cycle having the most negative total cost is selected to be canceled, which causes the maximum decrease of the objective function per unit flow augmentation. Computational studies showed that this rule usually results in fewer iterations than other strategies.

- **First eligible arc pivot rule**. At each iteration this method selects the first eligible arc. The method examines the arcs cyclically by starting each search process at the position where the previous eligible arc is found. If we reach the end of the arc list, the examination is continued from the beginning of the list again. If a pivot operation examines all non-tree arcs without finding an eligible arc, the solution is optimal and the algorithm terminates.

- **Block search pivot rule [10] .**This method cyclically examines certain subsets (blocks) of the arcs and selects the best eligible candidate among these arcs at each iteration. The block size B is an important parameter of this method. In fact, the previous two rules are special cases of this one with B = m and B = 1respectively.

- **The Candidate list pivot rule [14] .** In a so-called major iteration, this method examines the arcs cyclically to build a list containing at most L eligible arcs. This list is then used by at most K subsequent iterations to select an arc of maximum violation among the candidates.

If an arc becomes non-eligible, it is removed from the list.

Comparison:

The experimental study is available in [4] .

Pivot Rule	Aspects	Overall Performance
Dantzig`s rule	Easy method, usually performs iterations less than other rules. It calculates the reduced costs for all non tree arcs to select the entering arc at each iteration.	Poor performance in practice.
First eligible arc pivot rule	Easy method, it finds the entering arc quickly at each iteration , but it requires a lot of iterations .	Poor performance in practice
Block search pivot rule	It considers the best arc of the block and advances to the next block . It decreases the number of degenerate pivots in practice . The block size is an important parameter in this method.	Robust and effective
Candidate list pivot rule	It is similar to the previous rule, but it considers the same block of arcs in several consecutive pivots.	Les robust and effective than the previousrule .

4. CONCLUSION

Minimum cost flow problem (M.C.F) is a framework for modeling and formulating many variants of network flow problems . It is difficult to find a numerical example that illustrates how network simplex algorithm (big-M) solves M.C.F problem in enough details , describing its properties , and how it exploits the network structure to achieve the optimal solution , because N.S.A designed for solving large problems with large data .A variant of M.C.F (single source shortest path problem) is solved by N.S.A (big-M) version , and it turns out that artificial initialization guarantees constructing of strongly feasible initial tree easily and quickly . Implementing N.S.A requires a method for selecting such an entering arc at each iteration , which is usually known as pivot rule . Four well known pivot rules ,are presented and compared , each one has its aspects , which determines the behavior of N.S.A and the solution speed .

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 4, Issue 3, pp: (1-9), Month: September - December 2017, Available at: www.noveltyjournals.com

REFERENCES

[1] D.J. Kelly, G.M. O'Neill, "The minimum cost flow problem and the network simplex solution method", Master's thesis, University College, Dublin, 1991,2,4,5,6,11,13,24,27-29,39-46.

[2] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, Network Flows: Theory, Algorithms, and Applications, Prentice-Hall, Inc., 1993, 309,310,313,408 –425 .

[3] E. Eroglu, " An Application of Network Simplex Method for Minimum Cost Flow Problems " BALKANJM 01:117-130 , October 2013 .

[4] Z.Király, and P. Kovács, “Efficient implementations of minimum-cost flow algorithms”, Acta Universitatis Sapientiae, Informatica, 4 (1) (2012) 67–118.

[5] D. P.Bertsekas, "Network Optimization: Continuous and Discrete Models", Athena Scientific, Belmont, MA,1998 ,170 - 195.

[6] G. Geranis ," A Dual network exterior point simplex-type algorithm for the minimum cost network flow problem ", PhD thesis , University of Macedonia, 2012 . 20-32 .

[7] J. D. Kramer ,"Min-Cost Multicommodity Network Flows: A Linear Case for the Convergence and Re optimization of Multiple Single Commodity Network Flows" , Master's thesis, North Carolina State University, 2009 .11-14, 20, 21.

[8] G.B. Dantzig, Linear Programming and Extensions, Princeton University Press, Princeton, NJ, 1963. 74, 91.

[9] R.S. Barr, F. Glover, andD. Klingman, Enhancements to spanning tree labeling procedures for network optimization, INFOR 17 (1979) 16-34.

[10] M.D. Grigoriadis, An efficient implementation of the network simplex method, Math. Program. Stud. 26 (1986) 83-111.

[11] Kantorovich, L.V. “Mathematical Methods in the Organization and Planning of Production”, translated in Management Science (1960), Volume: 6, p: 336-422, 1939.

[12] D.Bertsimas, and J.N. Tsitsiklis, “Introduction to Linear Optimization”, Athena Scientific, 1997.

[13] J.B. Orlin. A polynomial time primal network simplex algorithm for minimum cost flows. Math. Program., 78:109-129, 1997.

[14] J.M. Mulvey. Pivot strategies for primal-simplex network codes. J. ACM 25 (1978) 266-270.