Conversion of Network Problem with Transfer Nodes, and Condition of Supplying the Demand of any Sink from the Particular Source to the Transportation Problem

F. Hamzezadeh, H. Salehi Fathabadi*

Received: 2 December 2014 ; Accepted: 13 May 2015

Abstract In this article we present an algorithm for converting a network problem with several sources and several sinks including several transfer nodes and condition of supplying the demand of any sink from a particular source to the transportation problem. Towards this end, and considering the very special structure of transportation algorithm, after implementing the shortest path algorithm or solving model 2 and determining the paths by the minimum cost, we let the algorithm to behave with any of these shortest paths as an arc. Although this problem is soluble by linear programming with network structure, but by converting it to transportation problem an efficient method may be proposed for solving it.

Keywords: Minimum cost, Network flow, Transportation problem, shortest path.

1 Introduction

Transportation problems are important linear programming problems which are applied in different fields. These problems are special modes of network flow problems. The first formula of transportation model, and its discussion, was developed by Hichcock[1]. Dantzig[2] applied simplex method for solving the transportation problem. Koopman[3] was the first who considered the relations between basic responses in transportation problems, and development of a tree. The issue of allocating determined budget for increasing the arcs’ capacity in transhipment problem was studied by Ahuja[4]. In this article we consider a flow network system with m source nodes and n sink nodes, including several transfer nodes. We interpret the m source nodes as the m producer and n sink nodes as the n customer. Suppose the policy of supplying demand of the j th customer has been designed in a manner that at least \( d_{ij} \) units of the product must be definitely supplied by the i th producer. Such problems are commonly applied in real life. For example, suppose m to be the number of producer of fuel material such as petrol, and n to be the number of country which needs this fuel. There is a contract between the producers and the countries such that the j th country must supply at least \( d_{ij} \) units of its required fuel from the i th producer. Accordingly, we present an algorithm for solving this problem. This paper has been arranged as follows: section 2 defines notations, primary

* Corresponding Author. (✉)
E-mail: hsalehi@ut.ac.ir (H. Salehi Fathabadi)

F. Hamzezadeh, H. Salehi Fathabadi* 
Department of Mathematics, College of Basic Science, Karaj Branch, Islamic Azad University, Alborz, Iran.
assumptions and mathematical model. Section 3 explains the algorithm. A numerical example will be considered in section 4.

2 Problem Statement

We consider a direct network with \( m \) source nodes, \( n \) sink nodes, several transfer nodes and a set of directed arcs, \( A \). We associate with each source node \( s_i, i = 1, \ldots, m \) an integer number \( a_i \), representing its supply. Also we associate with each sink node \( t_j, j = 1, \ldots, n \) an integer number \( b_j \), representing its demand. We suppose that \( a_i, b_j \geq 0 \). Each arc \( (p, q) \in A \) has an associated cost \( c'_{pq} \) that denotes the cost per unit flow on that arc. In addition, here \( d_{s_it_j} \) units of goods must be transferred from source \( s_i, i = 1, \ldots, m \) to sink \( t_j, j = 1, \ldots, n \). For example, we suppose \( m \) to be the number of producer of fuel material such as petrol, and \( n \) to be the number of country which needs this fuel. There is a contract between the producers and the countries such that the \( j \)th country must supply at least \( d_{s_it_j} \) units of its required fuel from the \( i \)th producer. Although this problem is soluble by linear programming with network structure, but by converting it to transportation problem an efficient method may be proposed for solving it.

2.1 Notation

We use the following definition throughout this paper:

\[ P_{s_it_j}^* : \text{The shortest path from source nodes } s_i, i = 1, \ldots, m \text{ to sink nodes } t_j, j = 1, \ldots, n \text{ with cost } \]
\[ c_{s_it_j}^* = \left\{ \sum_{(k,l)} c'_{kl} \left( k, l \right) \in P_{s_it_j}^* \right\} \]

\[ \bar{x}_{pq} : \text{Flow on the arc } (p, q) \in A . \]

\[ (s_i, t_j) : \text{The arc connecting } s_i, i = 1, \ldots, m \text{ and } t_j, j = 1, \ldots, n . \]

\[ x_{s_it_j} : \text{Flow from source } s_i, i = 1, \ldots, m \text{ to sink } t_j, j = 1, \ldots, n \text{ on the shortest arc from source } \]
\[ s_i, i = 1, \ldots, m \text{ to sink } t_j, j = 1, \ldots, n . \]

\[ d_{s_it_j} : \text{The amount of flow to be transhipped from source } s_i, i = 1, \ldots, m \text{ to sink } t_j, j = 1, \ldots, n . \]

The network flow satisfies the following assumptions [5,6].

i. The problem is balanced, that is, the total supply equals to total demand.

ii. Flow in the network must satisfy the so-called “flow conservation law”. This means that, except the source and sinks nodes; the input flow to a node equals the outflow of that node.

iii. Each node is perfectly reliable. This means that the amount of flow does not changed when passing through a node.
Now using the above notations and assumptions the mathematical model of the problem is constructed as:

\[
\min z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{s_{i}t_{j}} x_{s_{i}t_{j}} \quad (2.1)
\]

\[
s.t.
\]

\[
x_{s_{i}t_{j}} \geq d_{s_{i}t_{j}} \quad (2.2)
\]

\[
\sum_{j=1}^{n} x_{s_{i}t_{j}} \leq a_{i} \quad i = 1, \ldots, m \quad (2.3)
\]

\[
\sum_{i=1}^{m} x_{s_{i}t_{j}} \geq b_{j} \quad j = 1, \ldots, n \quad (2.4)
\]

\[
x_{s_{i}t_{j}} \geq 0
\]

Constraint (2.2) shows the flow from source nodes \( s_{i}, i = 1, \ldots, m \) to sink nodes \( t_{j}, j = 1, \ldots, n \) must be at least \( d_{s_{i}t_{j}} \). Constraint (2.3) ensures that the flow sent from source nodes \( s_{i}, i = 1, \ldots, m \) is not more than the supply of these source nodes; and constraint (2.4) ensures that the flow arrived at the sink nodes \( t_{j}, j = 1, \ldots, n \) is not less than the demand of these customers.

### 3 Algorithm

Consider Figure 1 and suppose there are \( n_{s_{i}t_{j}} \) paths from source nodes \( s_{i}, i = 1, \ldots, m \) to sink nodes \( t_{j}, j = 1, \ldots, n \). primarily, the shortest path algorithm is implemented. Now, without violating the problem’s totalities, we consider any of these shortest paths \( (P_{s_{i}t_{j}}^{*}) \) as a \( (s_{i}, t_{j}) \) arc with cost \( c_{s_{i}t_{j}} \). Now we apply the following steps.

![Network Flow Diagram](image)

**Fig. 1** Network flow

Step1. Implement the shortest path algorithm for any source \( s_{i}, i = 1, \ldots, m \) to sink \( t_{j}, j = 1, \ldots, n \), and call these shortest paths as \( P_{s_{i}t_{j}}^{*} \) with cost \( c_{s_{i}t_{j}} \).
Step 2. Consider any of these shortest paths from source nodes $s_i, i = 1, ..., m$ to sink nodes $t_j, j = 1, ..., n$ as a shortest arc $(s_i, t_j)$.

Step 3. Send $d_{s_i, t_j}$ unit of flow on this shortest arc from source node $s_i, i = 1, ..., m$ to sink $t_j, j = 1, ..., n$.

Step 4. Solve the transportation algorithm using cost $c_{s_i, t_j}, a_i - \sum_{j=1}^{n} d_{s_i, t_j}, i = 1, ..., m$ supply and $b_j - \sum_{i=1}^{m} d_{s_i, t_j}, j = 1, ..., n$ demand, and then find the optimal solution.

### 3.1 Another Method

In this section, we propose another method for solving this problem that is using the following model instead of steps 1 and 2. This model directly calculates the minimum cost from each source node $s_i, i = 1, ..., m$ to each sink node $t_j, j = 1, ..., n$. We consider this model for a network like the one given in fig.1, which has two sets of transfer nodes of $k = 1, ..., n_1$ and $l = 1, ..., n_2$. This model may be generalized to larger networks.

$$
\text{min } c_{s_i, t_j} \sum_{k=1}^{n_1} c_{s_i, k} \bar{x}_{s_i, k} + \sum_{k=1}^{n_1} \sum_{l=1}^{n_2} c_{k, l} x_{k, l} + \sum_{l=1}^{n_2} c_{j, t_{j}} x_{j, t_{j}} \lambda_{s_i, t_{j}}^{n} \quad i = 1, ..., m, \quad j = 1, ..., n
$$

s.t.

- $\sum_{k=1}^{n_1} \lambda_{s_i, k} = 1 \quad (3-1)$
- $\sum_{k=1}^{n_1} \lambda_{s_i, k} = 1 \quad (3-2)$
- $\sum_{l=1}^{n_2} \lambda_{k, l} = 1 \quad (3-3)$
- $\sum_{l=1}^{n_2} \lambda_{k, l} = 1 \quad (3-4)$
- $\sum_{k=1}^{n_1} \bar{x}_{s_i, k} \leq a_i \quad i = 1, ..., m \quad (3-5)$
- $\sum_{j=1}^{n_2} \bar{x}_{j, t_{j}} \geq b_j \quad j = 1, ..., n \quad (3-6)$

- $\lambda_{s_i, k} \in \{0,1\} \quad \text{for all } (s_i, k) \in A$
- $\lambda_{k, l} \in \{0,1\} \quad \text{for all } (k, l) \in A$
- $\lambda_{t_{j}, j}^{n} \in \{0,1\} \quad \text{for all } (l, t_{j}) \in A$
- $\bar{x}_{pq} \geq 0 \quad \text{for all } (p, q) \in A$
Condition (3-1) ensures that the model chooses only one arc from each source node to any of the first set transfer nodes; and conditions (3-2) to (3-4) act in a similar way as well. After solving this model we may observe that the optimal solution of this model is as the same as the solution resulted from the shortest path algorithm.

4 Numerical Example

We apply the small network shown in figure 2 for displaying the algorithm. This network includes 3 source nodes, 4 sink nodes and 4 medium nodes. Source nodes embraces supply/goods \((a_1,a_2,a_3)=(100,150,170)\), and sink nodes embrace demand \((b_1,b_2,b_3,b_4)=(150,100,50,120)\).

In addition, at least 20 units of goods (flow) from source 1\((s_1)\) to sink 2\((t_2)\), at least 25 units of goods (flow) from source 2\((s_2)\) to sink 1\((t_1)\), at least 15 units of goods (flow) from source 2\((s_2)\) to sink 4\((t_4)\) and at least 30 units of goods (flow) from source 3\((s_3)\) to sink 3\((t_3)\) must be transhipped. The transportation cost of any unit of goods (flow) has been shown on each arc.

Step1. We run the shortest path algorithm. The shortest path from source node 1 to sink node 1 is \(s_1 \rightarrow 1 \rightarrow 3 \rightarrow t_1\), ..., the shortest path from source node 3 to sink node 4 is \(s_3 \rightarrow t_4\) with cost \(c_{s_1t_1}=10\), ..., \(c_{s_3t_4}=5\).

Step2. \(P^{*}_{s_1t_1} \rightarrow (s_1,t_1)\), ..., \(P^{*}_{s_3t_4} \rightarrow (s_3,t_4)\).

Step3. We transfer 20 units of goods (flow) from source 1\((s_1)\) to sink 2\((t_2)\), ..., and 15 units of goods (flow) from source 2\((s_2)\) to sink 4\((t_4)\).

Step4. We run the transportation algorithm with using information and primary solution in Table 1. The optimal solutions are \(x_{11}=80\), \(x_{21}=10\), \(x_{22}=80\), \(x_{23}=20\), \(x_{31}=35\), \(x_{33}=105\) with cost 2995.

<table>
<thead>
<tr>
<th>Table 1 Table of transportation</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="" /></td>
</tr>
</tbody>
</table>
5 Conclusions

In this article we presented an algorithm which converts a network with several source nodes, several sink nodes and several transfer nodes, with condition of supplying a part of demand of any sink from a particular source, to the transportation problem, and then solves the problem by using transportation algorithm. Although this problem is soluble through flow networks, yet the method presented in this article is an efficient and considerable solution.

References