Network flow problem

From optimization

Authors: Blake Alexander, Aaron Frank, and Joshua Lee (ChE 345 Spring 2014)

Contents

Introduction

Network Flow Optimization problems form the most special class of linear programming problems. Transportation, electric, and communication networks are clearly common applications of Network Optimization. These types of problems can be viewed as minimizing transportation problems. This Network problem will include cost of moving materials through a network involving varying demands, parameters, and constraints depending on the locations that the materials are being brought to. Problems of these type are characterize Network Flow Optimization. The consideration of the connections between different parts of the Network is what makes these problems difficult, but extremely important and applicable.

A network consists of two types of of objects, which are, nodes and arcs (1). Node sets will be denoted by “N” with m being the number of nodes. These nodes are connected by arcs. These arcs are defined and direct, meaning that the arc that connects nodes 1 and 2 is not the same as the arc that connects nodes 3 and 4. It is therefore intuitive to denote arcs by their direction [i.e. arc (1,2), arc (3,4), (i.j)]. Knowing this, we can denote the set of all arcs in the network with “A.” The following expression is the subset of the set of all possible arcs (1):

$$A \subset (i, j); i, j \in N, i \neq j$$

The pair (N, A) is called a network. To specify a network flow problem, we need to specify the supply/demand of a material into a node. So for each i $\in$ N, let b$_i$ be the supply/demand to the node to the network at Node *i*. Depending on whether the amount of material moved to each node is negative or positive differentiates supply or demand. Thus, one can assume (1):

$$\sum_i b_i = 0 \text{ for } i \in N$$

To guide the solver in solving the paths, we assume that for each arc, there is an associated cost c $_{i\ j}$, for moving material. Furthermore, x $_{i\ j}$ is the quantity shipped down a certain arc.

With this information, the objective of the network flow problem is simple. The objective, or problem, is minimizing total cost of moving supplies while meeting demands (1):

$$\text{minimize} \sum c_i x_i \text{ for } (i,j) \in A$$

As stated earlier, Network Flow Optimization problems are limited by constraints. Depending on the circumstances of the problem, these constraints can have some variation.

Theorems

Network Flow problems have several theorems that are applied in different scenarios and circumstances to categorize questions. In turn, these problems become easier to solve with the following theorems.

Integrality Theorem

Integrality Theorem: For network flow problems with integer data, every basic feasible solution and, in particular, every basic optimal solution assigns integer flow to every arc. (1)

The Integrality Theorem is extremely applicable to the real world because often times network flow problems have integral supplies/demands and these problems require integer solutions. This restriction is generally applied when one is shipping indivisible units through a network (i.e. you wouldn’t ship ⅓ of a car from the assembly to the dealership). Solving such a network that follows the Integrality Theorem is quite simple. One can efficiently solve the problem using the simplex method to compute a basic optimal solution that is an integer.

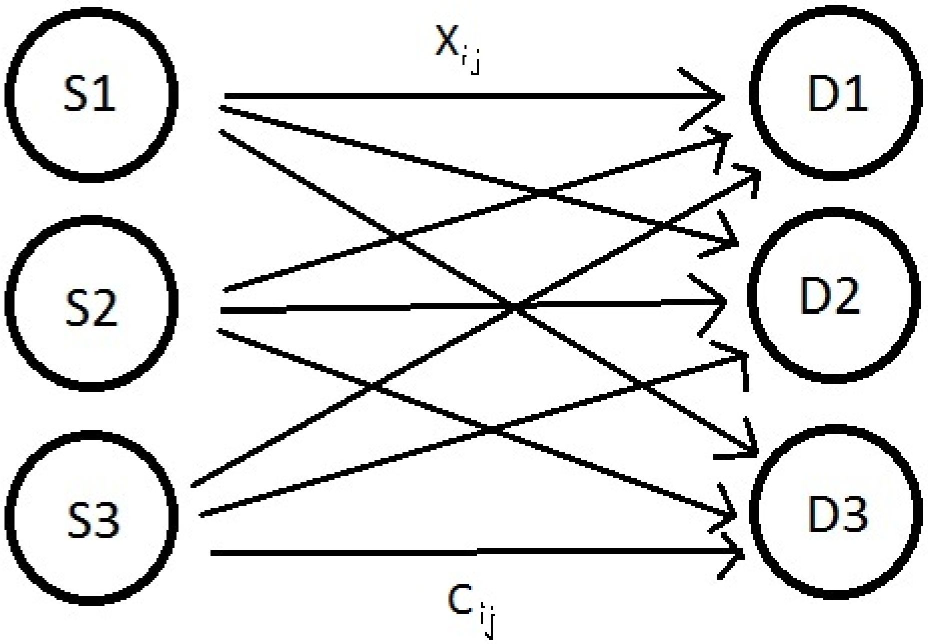
Tree Theorem

Tree Theorem: A square submatrix of “A˜ is a basis if and only if the arcs to which its columns correspond form a spanning tree.

The Tree Theorem is the most important theorem of Network Flow problems.

General Case

A basic example of the Network Flow Optimization problem is one based around transportation. There are three source nodes denoted S1, S2, and S3, and three demand nodes denoted D1, D2, and D3. Each source node can deliver its product to any demand node, and overall all products produced are consumed by the demand nodes. Each supply node has a different amount of product it can produce, and each demand node requires a different amount of product. The shipping costs from each supply node to each demand node are different which gives rise to how to distribute products so that demand is met at the lowest cost.



Strategies to Solve the Problem

$$\text{minimize} \sum_{i \in S} \sum_{j \in D} c_{ij} x_{ij} \quad \text{(A)}$$

$$\text{s.t.} \sum x_{ij} \leq S_i \quad \text{For All } i \in S \quad \text{(B)}$$

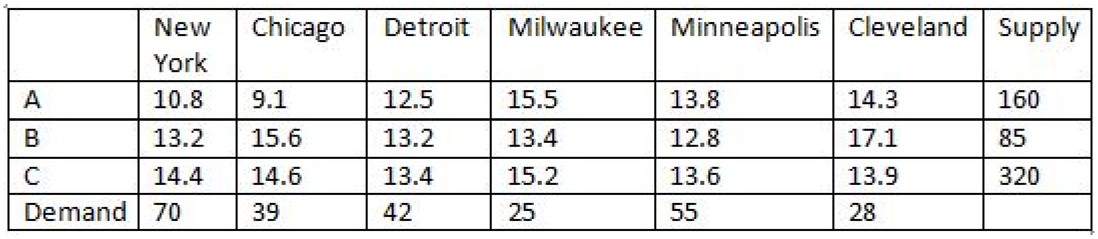
$$\sum x_{ij} \geq D_j \quad \text{For All } j \in D \quad \text{(C)}$$

$$x_{ij} \geq 0 \quad \text{For All } i \in S, i \in D \quad \text{(D)}$$

Equation (A) is the minimization of the product of cost and amount of product, which gives the total price. The total price is subject to the constraints of equations (B) and (C), where (B) is the condition that the sum of the products transported from source to demand is not greater than the total supply. Equation (C) is the constraint that the demand is not greater than the total amount of product shipped. Constraint (D) is to ensure that there is indeed a product shipped since otherwise all minimization problems would have an answer of zero. This situation can be solved by a software program such as GAMS.

Example

In a transportation example, the cost of the transportation is to be minimized given supply and demand constraints. The table below shows six cities with demand constraints and three factories (A,B, and C) with maximum supply constraints. Also included are the transportation costs between each factory and city.



	New York	Chicago	Detroit	Milwaukee	Minneapolis	Cleveland	Supply
A	10.8	9.1	12.5	15.5	13.8	14.3	160
B	13.2	15.6	13.2	13.4	12.8	17.1	85
C	14.4	14.6	13.4	15.2	13.6	13.9	320
Demand	70	39	42	25	55	28	

By running a linear optimization program in GAMS to minimize transportation cost, the minimum cost of $3,064.10 was determined. A portion of the GAMS code is shown below: variable x(i,j), z;

```
equations cost, sup(i), dem(j);
```

```
cost..   z=e= sum((i,j), c(i,j)*x(i,j) );
sup(i)..   sum(j, x(i,j) ) =l= a(i)    ;
dem(j)..   sum(i, x(i,j) ) =g= d(j)    ;
```

```
x.lo(i,j) = 0;
```

```
model transport /all/;
solve transport using lp minimizing z;
```

All *i* refers to the plants A, B, and C, and all *j* refers to the six markets. The cost is minimized according to the supply and demand constraints given. Since this is a linear problem the lp solver can be used.

Conclusion

Network Flow problems are useful for minimizing different issues that arise when considering many different situations, but especially transportation, electric, and communications problems. The complex connections between nodes and arcs can be applied to problems of varying magnitude.

Sources

(1) R.J. Vanderbei, Linear Programming: Foundations and Extensions. Springer, 2008.

Steward: Dajun Yue, Fengqi You

Date Presented: Apr. 10, 2014

Retrieved from "https://optimization.mccormick.northwestern.edu/index.php?title=Network_flow_problem&oldid=887"

- This page was last modified on 4 June 2014, at 15:52.
- This page has been accessed 22,055 times.