Department of Industrial Engineering & Operations Research

IEOR 162 Linear Programming & Network Flows (Spring 2022)

1 The general network flow problem

A common scenario of a network-flow problem arising in industrial logistics concerns the distribution of a single homogenous product from plants (origins) to consumer markets (destinations). The total number of units produced at each plant and the total number of units required at each market are assumed to be known. The product need not be sent directly from source to destination, but may be routed through intermediary points reflecting warehouses or distribution centers. Further, there may be capacity restrictions that limit some of the shipping links. The objective is to minimize the variable cost of producing and shipping the products to meet the consumer demand.

The sources, destinations, and intermediate points are collectively called *nodes* of the network, and the transportation links connecting nodes are termed *arcs*. Although a production/distribution problem has been given as the motivating scenario, there are many other applications of the general model. Table 8.1 indicates a few of the many possible alternatives.

Table 8.1 Examples of Network Flow Problems

	Urban transportation	*Communication systems*	*Water resources*
Product	Buses, autos, etc.	Messages	Water
Nodes	Bus stops, street intersections	Communication centers, relay stations	Lakes, reservoirs, pumping stations
Arcs	Streets (lanes)	Communication channels	Pipelines, canals, rivers

A numerical example of a network-flow problem is given in Figure 8.1. The nodes are represented by numbered circles and the arcs by arrows. The arcs are assumed to be *directed* so that, for instance, material can be sent from node 1 to node 2, but not from node 2 to node 1. Generic arcs will be denoted by (i, j), so that $(4, 5)$ means the arc *from* 4 *to* 5. Note that some pairs of nodes, for example 1 and 5, are not connected directly by an arc.

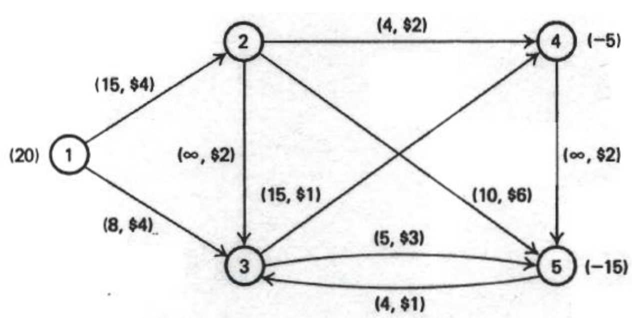

Fig. 8.1 Minimum-cost flow problem.

Figure 8.1 exhibits several additional characteristics of network flow problems. First, a flow capacity is assigned to each arc, and second, a per-unit cost is specified for shipping along each arc. There characteristics are shown next to each arc. This, the flow on arc 2-4 must be between 0 and 4 units, and each unit flow on this arc costs \$2.00. The ∞'s in the figure have been used to denote unlimited flow capacity on arcs $(2, 3)$ and $(4, 5)$. Finally, the numbers in parenthesis next to the nodes give the material supplied or demanded at that node. In this case, node 1 is an origin or *source node* supplying 20 units, and nodes 4 and 5 are destinations or *sink nodes* requiring 5 and 15 units, respectively, as indicated by the negative signs. The remaining nodes have no net supply or demand; they are intermediate points, often referred to as *transshipment nodes.*

The objective is to find the minimum-cost flow pattern to fulfil demands from the source nodes. Such problems usually are referred to as *minimum-cost flow* or *capacitated transhipment* problems. To transcribe the problem into a formal linear program, let

$$x_{ij} = \text{Number of units shipped from node } i, \text{ to } j \text{ using arc } (i,j).$$

Then the tabular form of the linear-programming formulation associated with the network of Fig. 8.1 is as shown in Table 8.2.

Table 8.2 Tableau for Minimum-Cost Flow Problem

	x_{12}	x_{13}	x_{23}	x_{24}	x_{25}	x_{34}	x_{35}	x_{45}	x_{53}	*Righthand side*
Node 1	1	1								20
Node 2	−1		1	1	1					0
Node 3		−1	−1			1	1		−1	0
Node 4				−1		−1		1		−5
Node 5					−1		−1	−1	1	−15
Capacities	15	8	∞	4	10	15	5	∞	4	
Objective function	4	4	2	2	6	1	3	2	1	(Min)

The first five equations are flow-balance equations at the nodes. They state the conservation-of-flow law,

$$\begin{pmatrix}\text{Flow out}\\ \text{of a node}\end{pmatrix} - \begin{pmatrix}\text{Flow into}\\ \text{a node}\end{pmatrix} = \begin{pmatrix}\text{Net supply}\\ \text{at a node}\end{pmatrix}.$$

It is important to recognize the special structure of these balance equations. Note that there is one balance equation for each node in the network. The flow variables x_{ij} have only 0, +1, and −1 coefficients in these equations. Further, each variable appears in exactly two balance equations, once with a +1 coefficient, corresponding to the node from which the arc emanates; and once with a −1 coefficient, corresponding to the node upon which the arc is incident. This type of tableau is referred to as a *node-arc incident matrix*; it completely describes the physical layout of the network. It is this particular structure that we shall exploit developing specialized, efficient algorithms.

The remaining two rows in the table give the upper bounds on the variables and the cost of sending one unit of flow across an arc. For example, x_{12} is constrained by $0 \leq x_{12} \leq 15$ and appears in the objective function as $2x_{12}$. In this example the lower bounds on the variables are taken implicitly to be zero, although in general there may be nonzero lower bounds.

This example is an illustration of the following general *minimum-cost flow* problem with n nodes:

$$\text{Minimize } z = \sum_i \sum_j c_{ij} x_{ij},$$

subject to:

$$\sum_j x_{ij} - \sum_k x_{ki} = b_i \quad (i = 1, 2, \ldots, n), \quad \text{[Flow balance]}$$

$$\ell_{ij} \leqq x_{ij} \leqq u_{ij}. \qquad \text{[Flow capacities]}$$