8.4 TRANSPORT NETWORKS

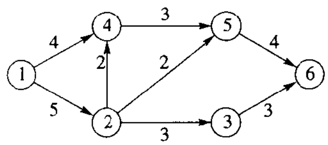

Figure 8.69

We have previously examined several uses of labeled graphs. In this section we return to the idea of a directed graph (digraph). An important use of labeled digraphs is to model what are commonly called transport networks. Consider the labeled digraph shown in Figure 8.69. This might represent a pipeline that carries water from vertex 1 to vertex 6 as part of a municipal water system. The label on an edge represents the maximum flow that can be passed through that edge and is called the **capacity** of the edge. Many situations can be modeled in this way For instance, Figure 8.69 might as easily represent an oil pipeline, a highway system, a communications network, or an electric power grid. The vertices of a network are usually called nodes and may denote pumping stations, shipping depots, relay stations, or highway interchanges.

More formally, a **transport network**, or a **network**, is a connected digraph N with the following properties:

(a) There is a unique node, the **source**, that has in-degree 0. We generally label the source node 1.

(b) There is a unique node, the **sink**, that has out-degree 0. If N has n nodes, we generally label the sink as node n.

(c) The graph N is labeled. The label, C_{ij}, on edge (i, j) is a nonnegative number called the capacity of the edge.

For simplicity we also assume that all edges carry material in one direction only; that is, if (i, j) is in N, then (j, i) is not.

Flows

The purpose of a network is to implement a flow of water, oil, electricity, traffic, or whatever the network is designed to carry. Mathematically, a **flow** in a network N is a function that assigns to each edge (i, j) of N a nonnegative number F_{ij} that does not exceed C_{ij}. Intuitively, F_{ij} represents the amount of material passing through the edge (i, j) when the flow is F. Informally, we refer to F_{ij} as the flow through edge (i, j). We also require that for each node other than the source and sink, the sum of the F_{ik} on edges entering node k must be equal to the sum of the F_{kj} on edges leaving node k. This means that material cannot accumulate, be created, dissipate, or be lost at any node other than the source or the sink. This is called **conservation of flow**. A consequence of this requirement is that the sum of the flows leaving the source must equal the sum of the flows entering the sink. This sum is called the **value of the flow**, written $value(F)$. We can represent a flow F by labeling each edge (i, j) with the pair (C_{ij}, F_{ij}). A flow F in the network represented by Figure 8.69 is shown in Figure 8.70.

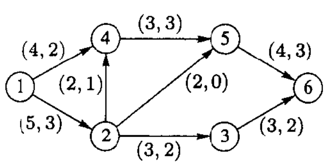

Figure 8.70

EXAMPLE 1 In Figure 8.70, flow is conserved at node 4 since there are input flows of size 2 and 1, and an output flow of size 3. (Verify that flow is conserved properly at the other nodes.) Here $value(F) = 5$. ∎

Maximum Flows

For any network an important problem is to determine the maximum value of a flow through the network and to describe a flow that has the maximum value. For obvious reasons this is commonly referred to the maximum flow problem.

EXAMPLE 2 Figure 8.71(a) shows a flow that has value 8. Three of the five edges are carrying their maximum capacity. This seems to be a good flow function, but Figure 8.71(b) shows a flow with value 10 for the same network. ■

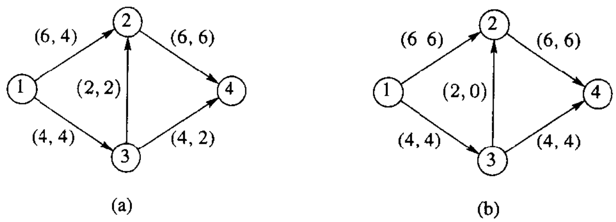

Figure 8 71

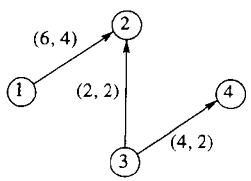

Figure 8.72

Example 2 shows that even for a small network, we need a systematic procedure for solving the maximum flow problem. Examining the flow in Figure 8.71(a) shows that using the edge from node 3 to node 2 as we did was a mistake. We should reduce flow in edge (3, 2) so that we can increase it in other edges.

Suppose that in some network N we have an edge (i, j) that is carrying a flow of 5 units. If we want to reduce this flow to 3 units, we can imagine that it is combined with a flow of two units in the opposite direction. Although edge (j, i) is not in N, there is no harm in considering such a *virtual flow* as long as it only has the effect of reducing the existing flow in the actual edge (i, j). Figure 8.72 displays a portion of the flow shown in Figure 8.71(a).

The path π : 1, 2, 3, 4 is not an actual path in this network, since (2, 3) is not an actual edge. However, π is a path in the symmetric closure of the network. (Refer to Section 4.7 for symmetric closure.) Moreover, if we consider a virtual flow of two units through π, the effect on the network is to increase the flows through edges (1, 2) and (3, 4) by two units and decrease the flow through edge (3, 2) by two units. Thus, the flow of Figure 8.71(a) becomes the flow of Figure 8.71(b).

We now describe this improvement in general terms. Let N be a network and let G be the symmetric closure of N. Choose a path in G and an edge (i, j) in this path If (i, j) belongs to N, then we say this edge has positive excess capacity if $e_{ij} = C_{ij} - F_{ij} > 0$. If (i, j) is not an edge of N, then we are traveling this edge in the wrong direction. In this case we say (i, j) has excess capacity $e_{ij} = F_{ji}$ if $F_{ji} > 0$. Then increasing flow through edge (i, j) will have the effect of reducing F_{ji}. We now give a procedure for solving a maximum flow problem.

A Maximum Flow Algorithm

The algorithm we present is due to Ford and Fulkerson and is often called the **labeling algorithm**. The labeling referred to is an additional labeling of nodes. We have used integer capacities for simplicity, but Ford and Fulkerson show that this algorithm will stop in a finite number of steps if the capacities are rational numbers.

Let N be a network with n nodes and G be the symmetric closure of N All edges and paths used are in G. Begin with all flows set to 0 As we proceed, it will be convenient to track the excess capacities in the edges and how they change rather than tracking the increasing flows. When the algorithm terminates, it is easy to find the maximum flow from the final excess capacities.

◆ THE LABELING ALGORITHM

Step 1 Let N_1 be the set of all nodes connected to the source by an edge with positive excess capacity. Label each j in N_1 with $[E_j, 1]$, where E_j is the excess capacity e_{1j} of edge $(1, j)$. The 1 in the label indicates that j is connected to the source, node 1.

Step 2 Let node j in N_1 be the node with smallest node number and let $N_2(j)$ be the set of all unlabeled nodes, other than the source, that are joined to node j and have positive excess capacity. Suppose that node k is in $N_2(j)$ and (j, k) is the edge with positive excess capacity. Label node k with $[E_k, j]$, where E_k is the minimum of E_j and the excess capacity e_{jk} of edge (j, k). When all the nodes in $N_2(j)$ are labeled in this way, repeat this process for the other nodes in N_1. Let $N_2 = \bigcup_{j \in N_1} N_2(j)$.

Note that after Step 1, we have labeled each node j in N_1 with E_j, the amount of material that can flow from the source to j through one edge and with the information that this flow came from node 1. In Step 2, previously unlabeled nodes k that can be reached from the source by a path $\pi: 1, j, k$ are labeled with $[E_k, j]$. Here E_k is the maximum flow that can pass through π since it is the smaller of the amount that can reach j and the amount that can then pass on to k. Thus when Step 2 is finished, we have constructed two-step paths to all nodes in N_2. The label for each of these nodes records the total flow that can reach the node through the path and its immediate predecessor in the path. We attempt to continue this construction increasing the lengths of the paths until we reach the sink (if possible). Then the total flow can be increased and we can retrace the path used for this increase.

Step 3 Repeat Step 2, labeling all previously unlabeled nodes N_3 that can be reached from a node in N_2 by an edge having positive excess capacity. Continue this process forming sets $N_4, N_5, \ldots$ until after a finite number of steps either

(i) the sink has not been labeled and no other nodes can be labeled. It can happen that no nodes have been labeled; remember that the source is not labeled.
or

(ii) the sink has been labeled.

Step 4 In case (i), the algorithm terminates and the total flow then is a maximum flow. (We show this later.)

Step 5 In case (ii) the sink, node n, has been labeled with $[E_n, m]$ where

E_n is the amount of extra flow that can be made to reach the sink through a path π. We examine π in reverse order. If edge $(i, j) \in N$, then we increase the flow in (i, j) by E_n and decrease the excess capacity e_{ij} by the same amount. Simultaneously, we increase the excess capacity of the (virtual) edge (j, i) by E_n since there is that much more flow in (i, j) to reverse. If, on the other hand, $(i, j) \notin N$, we decrease the flow in (j, i) by E_n and increase its excess capacity by E_n. We simultaneously decrease the excess capacity in (i, j) by the same amount, since there is less flow in (i, j) to reverse. We now have a new flow that is E_n units greater than before and we return to Step 1.

EXAMPLE 3 Use the labeling algorithm to find a maximum flow for the network in Figure 8.69.

Solution Figure 8.73 shows the network with initial capacities of all edges in G. The initial flow in all edges is zero.

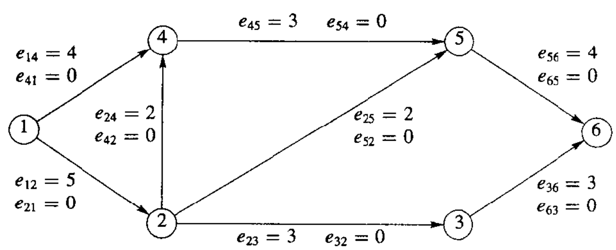

Figure 8.73

STEP 1: Starting at the source, we can reach nodes 2 and 4 by edges having excess capacity, so $N_1 = \{2, 4\}$. We label nodes 2 and 4 with the labels [5, 1] and [4, 1], respectively, as shown in Figure 8.74

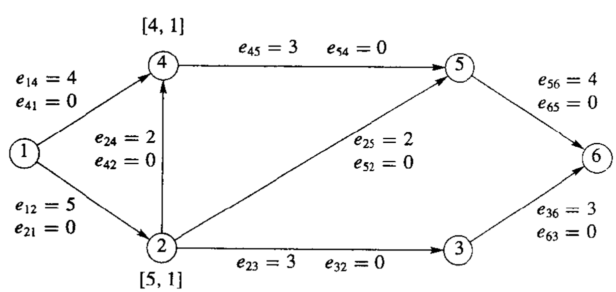

Figure 8.74

STEP 2: From node 2 we can reach nodes 5 and 3 using edges with positive excess capacity. Node 5 is labeled with [2, 2] since only two additional units of flow can pass through edge (2, 5). Node 3 is labeled with [3, 2] since only

3 additional units of flow can pass through edge (2, 3). The result of this step is shown in Figure 8.75.

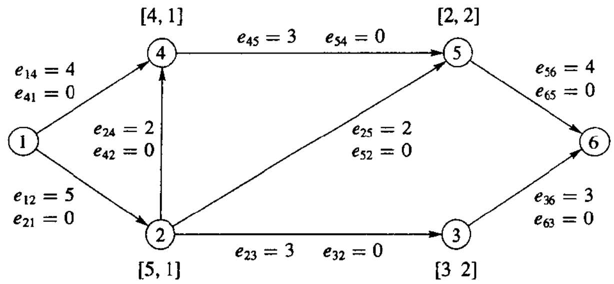

Figure 8.75

We cannot travel from node 4 to any unlabeled node by one edge. Thus, $N_2 = \{3, 5\}$ and Step 2 is complete.

***STEP 3*:** We repeat Step 2 using N_2. We can reach the sink from node 3 and 3 units through edge (3, 6). Thus the sink is labeled with [3, 3]

***STEP 5*:** We work backward through the path 1, 2, 3, 6 and subtract 3 from the excess capacity of each edge, indicating an increased flow through that edge, and adding an equal amount to the excess capacities of the (virtual) edges. We now return to Step 1 with the situation shown in Figure 8.76.

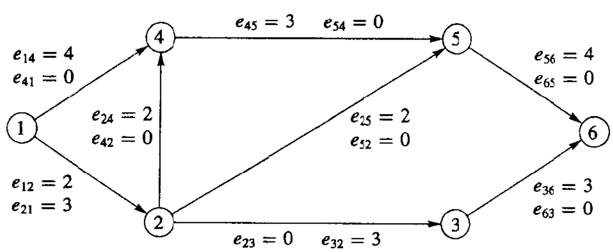

Figure 8.76

Proceeding as before, nodes 2 and 4 are labeled [2, 1] and [4, 1], respectively. Note that E_2 is now only 2 units, the new excess capacity of edge (1, 2). Node 2 can no longer be used to label node 3, since there is no excess capacity in the edge (2, 3). But node 5 now will be labeled [2, 2]. Once again no unlabeled node can be reached from node 4, so we move to Step 3. Here we can reach node 6 from node 5 so node 6 is labeled with [2, 5]. The final result of Step 3 is shown in Figure 8.77, and we have increased the flow by 2 units to a total of 5 units.

We move to Step 5 again and work back along the path 1, 2, 5, 6, subtracting 2 from the excess capacities of these edges and adding 2 to the capacities of the corresponding (virtual) edges. We return to Step 1 with Figure 8.78.

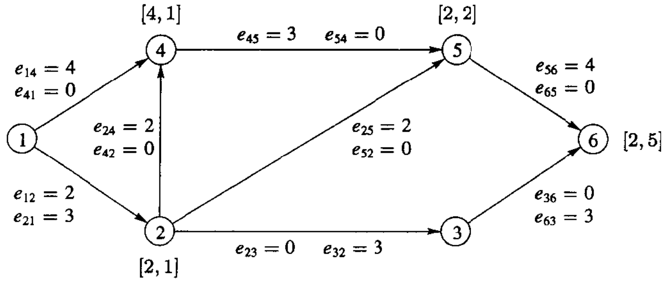

Figure 8.77

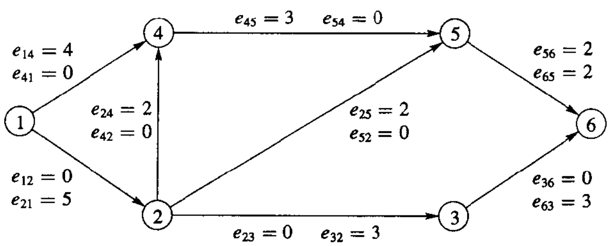

Figure 8.78

This time Steps 1 and 2 produce the following results. Only node 4 is labeled from node 1, with [4, 1]. Node 5 is the only node labeled from node 4, with [3, 4]. Step 3 begins with Figure 8.79.

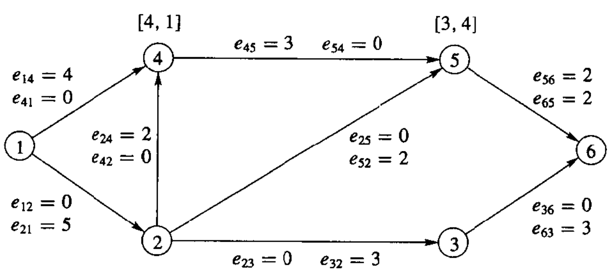

Figure 8.79

At this point, node 5 could label node 2 using the excess capacity of edge (5, 2). (Verify that this would label node 2 with [2, 5].) However, node 5 can also be used to label the sink. The sink is labeled [2, 5] and the total flow is increased to 7 units. In Step 5, we work back along the path 1, 4, 5, 6, adjusting excess capacities. We return to Step 1 with the configuration shown in Figure 8.80.

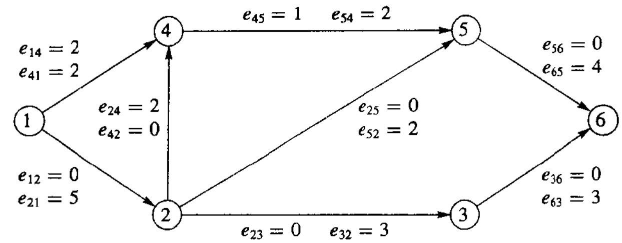

Figure 8.80

Verify that after Steps 1, 2, and 3, nodes 4, 5, and 2 have been labeled as shown in Figure 8.81 and no further labeling is possible. The final labeling of node 2 uses the virtual edge (5, 2).

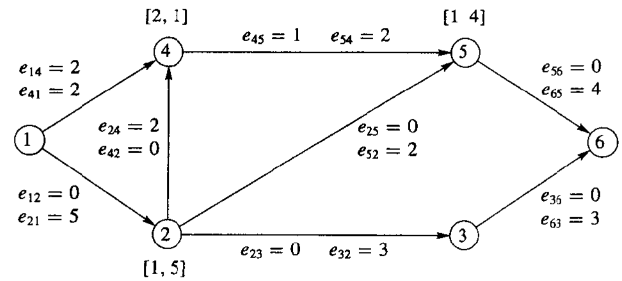

Figure 8.81

Thus, the final overall flow has value 7. By subtracting the final excess capacity e_{ij} of each edge (i, j) in N from the capacity C_{ij}, the flow F that produces the maximum value 7 can be seen in Figure 8.82. ■

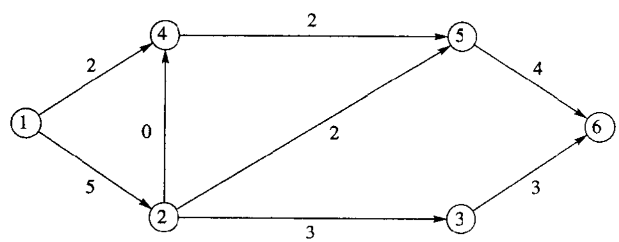

Figure 8.82

There remains the problem of showing that the labeling algorithm produces a maximum flow. First, we define a **cut** in a network N as a set K of edges having the property that every path from the source to the sink contains at least one edge from K. In effect, a cut does "cut" a digraph into two pieces, one containing the source and one containing the sink. If the edges of a cut were removed, *nothing* could flow from the source to the sink. The **capacity of a cut K**, $c(K)$, is the sum of the capacities of all edges in K.

EXAMPLE 4 Figure 8.83 shows two cuts for the network given by Figure 8.69. Each cut is marked by a jagged line and consists of all edges touched by the jagged line. Verify that $c(K_1) = 10$ and $c(K_2) = 7$. ■

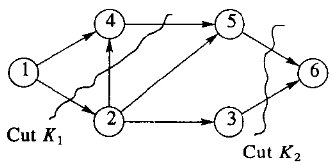

Figure 8.83

If F is any flow and K is any cut, then $value(F) \leq c(K)$. This is true because all parts of F must pass through the edges of K, and $c(K)$ is the maximum amount that can pass through the edges of K. Now suppose for some flow F and some cut K, $value(F) = c(K)$; in other words, the flow F uses the full capacity of all edges in K. Then F would be a flow with maximum value, since no flow can have value bigger than $c(K)$. Similarly, K must be a minimum capacity cut, because every cut must have capacity at least equal to $value(F)$. From this discussion we conclude the following.

Theorem 1 The Max Flow Min Cut Theorem A maximum flow F in a network has value equal to the capacity of a minimum cut of the network. ●

We now show that the labeling algorithm results in a maximum flow by finding a minimum cut whose capacity is equal to the value of the flow. Suppose that the algorithm has been run and has stopped at Step 4. Then the sink has not been labeled. Divide the nodes into two sets, M_1 and M_2, where M_1 contains the source and all nodes that have been labeled, and M_2 contains all unlabeled nodes, other than the source. Let K consists of all edges of the network N that connect a node in M_1 with a node in M_2. Any path π in N from the source to the sink begins with a node in M_1 and ends with a node in M_2. If i is the last node in π that belongs to M_1 and j is the node that follows i in the path, then j belongs to M_2 and so by definition (i, j) is in K. Therefore, K is a cut.

Now suppose that (i, j) is an edge in K, so that $i \in M_1$ and $j \in M_2$. The final flow F produced by the algorithm must result in (i, j) carrying its full capacity; otherwise, we could use node i and the excess capacity to label j, which by definition is not labeled. Thus the value of the final flow of the algorithm is equal to the capacity $c(K)$, and so F is a maximum flow.

EXAMPLE 5 The minimum cut corresponding to the maximum flow found in Example 3 is $K = \{(5, 6), (3, 6)\}$ with $c(K) = 7 = value(F)$ ■

8.4 Exercises

In Exercises 1 through 4 (Figures 8.84 through 8.87), label the network in the given figure with a flow that conserves flow at each node, except the source and the sink. Each edge is labeled with its maximum capacity.

1.

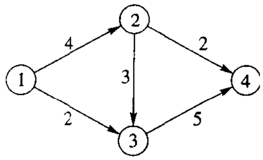

Figure 8.84

2.

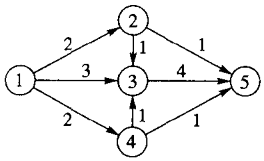

Figure 8.85

3.

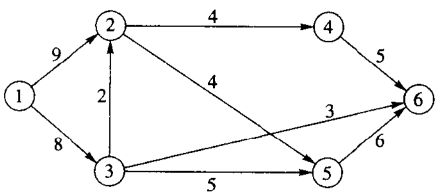

Figure 8.86

4.

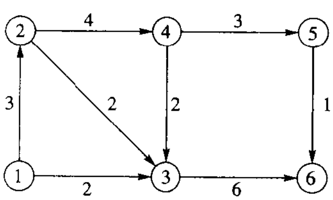

Figure 8.87

In Exercises 5 through 10, find a maximum flow in the given network by using the labeling algorithm.

5. The network shown in Figure 8.84
6. The network shown in Figure 8.85
7. The network shown in Figure 8.86
8. The network shown in Figure 8.87
9. The network shown in Figure 8 88

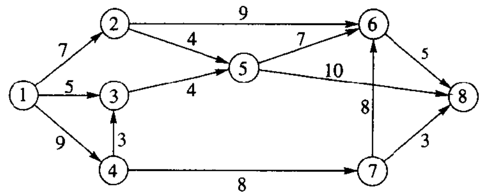

Figure 8.88

10. The network shown in Figure 8.89

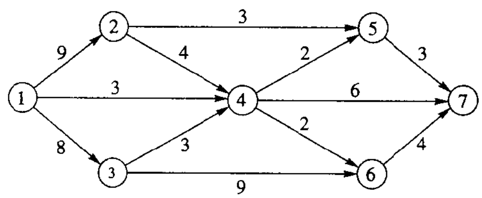

Figure 8.89

In Exercises 11 through 16, find the minimum cut that corresponds to the maximum flow for the given network.

11. The network of Exercise 5
12. The network of Exercise 6
13. The network of Exercise 7
14. The network of Exercise 8
15. The network of Exercise 9
16. The network of Exercise 10

8.5 MATCHING PROBLEMS

The definition of a transport network can be extended, and the concept of a maximal flow in a network can be used to model situations that, at first glance, do not seem to be network problems. We consider two examples in this section.

The first example is to allow a network to have multiple sources or multiple