Minimum Cost Flow Problems

MA428, Dr Katerina Papadaki

I. Introduction

In this lecture we will cover the following:

1) Problem definition
2) LP Formulation
3) Characteristics of Solutions
4) Network Simplex Algorithm
5) Integrality Theorems and Duality
6) Applications

The material covered in this lecture can be found in chapter 11 of AMO.

II. Problem definition

A. Preliminaries

Consider a network $G = (N, A)$ made up of $n = |N|$ nodes and $m = |A|$ arcs. We are concerned with transporting a single commodity over the network. Each arc (i, j) has a cost c_{ij} associated with it, which is the cost per unit of flow on the arc. For example if the flow on (i, j) is x_{ij}, then the total cost on the arc is $c_{ij}x_{ij}$. Also, each arc has a capacity u_{ij}, which is an upper bound on the flow on that arc. Let $b(i)$ denote the supply or demand of the commodity at each node $i \in N$. For $p > 0$, when $b(i) = +p$ we say that node i has a supply of p units, and when $b(i) = -p$ we say that node i has a demand of p units.

K. Papadaki is with the Department of Management, Operational Research Group at London School of Economics, United Kingdom

E-mail: k.p.papadaki@lse.ac.uk

Definition The problem of transporting the commodity through the network at a minimum total cost, while satisfying demand and supply conditions at nodes and capacity constraints on the arcs, is called the *minimum cost flow problem* (*min cost flow problem*).

Figure 1 shows an example of a min cost flow problem where we denote the supply/demand $b(i)$ on each node, and the cost and capacity c_{ij}, u_{ij} on each arc. Node we do not label nodes that have zero demand or supply.

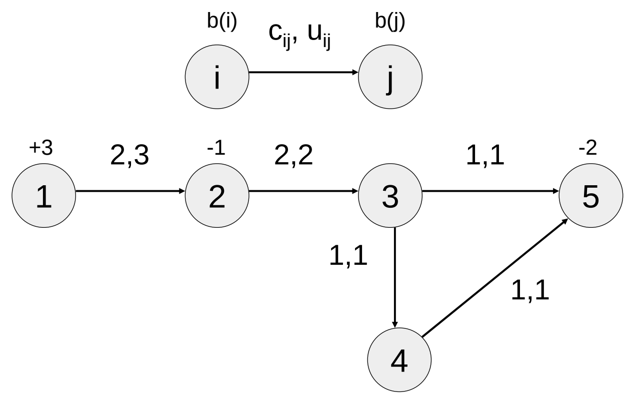

Fig. 1. An example of a min cost flow problem with non-zero supply/demand $b(i)$ on the nodes, and c_{ij}, u_{ij} on the arcs.

B. Assumptions

1) *Integral Data*: We assume that the costs, supplies, demands and capacities are integral. If the data are rational numbers we can always scale them to be integers by multiplying them by a constant. Note that costs can be scaled separately, but supplies, demands and capacities have to be scaled together since they depend on each other. When the integer solution is found we can divide the flows and the costs by the respective constants to find the optimal rational solution. If the data are irrational numbers, then this becomes problematic, even though in practice this is hardly ever the case.
2) *Directed Graph*: We assume that we solve the min cost flow problem on a directed graph (i.e. a network).

If the graph is not directed, then we can replace each edge (i, j) with arcs (i, j), (j, i). The original edge (i, j) had a cost of c_{ij} associated with each unit of flow on the edge in either direction. Thus we can associate the cost per unit of flow c_{ij} on each of the two arcs. Further, the original edge (i, j) had a capacity u_{ij} on the total flow that goes through the edge on both directions. If we let x_{ij} and x_{ji} the flow that goes through arcs (i, j) and (j, i) respectively and let x'_{ij} be the total flow that goes through edge (i, j), then $x'_{ij} = x_{ij} + x_{ji}$. Thus, the capacity of the edge is on the sum of the flows of the two arcs: $x_{ij} + x_{ji} \leq u_{ij}$. This problem can be easily solved by noting that in an optimal solution one of the flows x_{ij}, x_{ji} will have to be zero. This is because it is suboptimal to move the same commodity twice through the edge paying a cost every time.

3) *Feasibility 1: Supply equals Demand*: We assume that the sum of demands equals the sum of supplies.

   This means that: $\sum_{i \in N} b(i) = 0$. This assumption is necessary to have a feasible solution that satisfies all supplies and demands. However, it is not sufficient. We still need to check that capacity constraints allow for the demand and supplies to be satisfied (see assumption 4) below).

4) *Feasibility 2: Capacity allows for a feasible solution*: We assume that the capacity constraints allow for the demand and supplies to be satisfied.

   In Figure 2(a) there is no feasible solution since the capacity of 2 does not allow for the supply/demand of 3 units to pass through the network. We can check whether the minimum cost flow problem has a feasible solution by solving a max flow problem as follows. Create a source node s and a terminal node t. Create an arc (s, i) with a capacity $u_{si} = b(i)$ to every supply node i with $b(i) > 0$. Also create an arc (i, t) with a capacity $u_{it} = -b(i)$ from every demand node i with $b(i) < 0$. Now solve a max flow problem from s to t. If the max flow saturates every source arc (s, i) (or equivalently every terminal arc (i, t)), then $b(i)$ units of flow entered node i and thus $b(i)$ units left node i, which satisfies the supply of node i. Similarly, for each saturated arc (i, t), if $b(i)$ units of flow left node i that means that $b(i)$ units entered node i and satisfied its demand. Thus, the solution to this max flow problem gives us a feasible solution to the minimum cost flow problem. Figure 2(b) shows the reformulation of the min cost flow problem shown in Figure 2(a) to a max flow problem. As we can see from Figure 2(b), the max flow is 2 which does not saturate the arcs $(s, 1)$ and $(2, t)$. Thus, there is no feasible flow for the min cost flow problem.

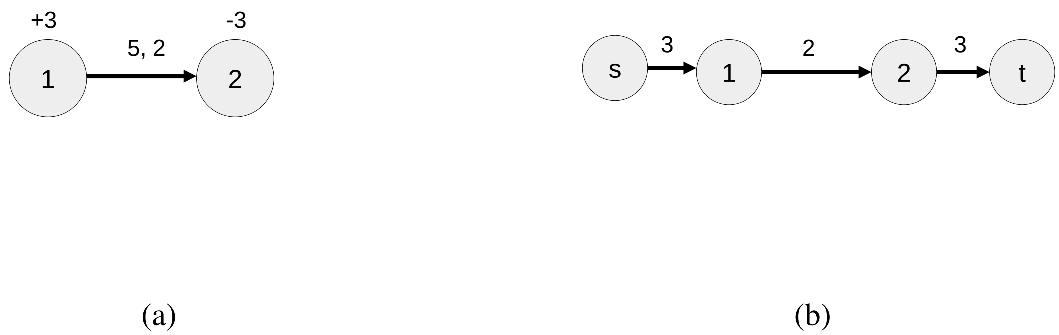

Fig. 2. (a) A min cost flow problem with an infeasible flow, where c_{ij}, u_{ij} is shown on the arc; (b) We formulate the problem as a max flow problem, where on each arc the capacities u_{ij} are shown. As we can see the max flow from s to t is 2 which does not saturate arcs $(s, 1)$ and $(2, t)$.

5) *Nonnegative Arc Costs*: We assume that $c_{ij} \geq 0$.

   This is a way of ensuring that there are no unbounded solutions. We can get an unbounded solution if and only if there is a negative cost directed cycle in the graph with all the arcs on the cycle having infinite capacity. So one could allow negative costs but forbid negative cycles with infinite capacity. Assuming $c_{ij} \geq 0$ is sufficient for our purposes.

6) *There is an uncapacitated directed path between every pair of nodes.*

   This assumption ensures that we can easily find a feasible solution. It is easy to satisfy the assumption by adding an artificial arc $(1, j)$ and $(j, 1)$ for each $j \in N$. If we assign a very large cost and an infinite capacity to all the artificial arcs, such an arc would never appear in the minimum cost solution unless the original problem has no feasible solution. When we do not know whether there is a feasible solution and we do not want to check by solving a max flow problem as we described in 4) above, we can add these artificial arcs.

III. LP Formulations

A. LP Formulation of the Minimum Cost Flow problem

Let x_{ij} be the amount of flow on arc (i, j). Then the minimum cost flow problem can be formulated as follows:

$$\begin{aligned} \min_{x_{ij}} \quad & \sum_{(i,j)\in A} c_{ij}x_{ij} \\ \text{s.t.} \sum_{j:(i,j)\in A} x_{ij} - & \sum_{k:(k,i)\in A} x_{ki} = b(i) \quad \text{for } i \in N \\ & 0 \le x_{ij} \le u_{ij} \quad \text{for all } (i,j) \in A \end{aligned} \tag{1}$$

The above formulation minimizes the total cost of all flows. The equation constraints ensure that the demands and supplies are satisfied at each node and the inequality constraints ensure that the flows are positive and within capacity.

B. Comparison with Shortest Path and Maximum Flow formulations

The shortest path problem and the maximum flow problem are special cases of the min cost flow problem.

To see that let's remind ourselves of the shortest path formulation. We let x_{ij} be number of shortest paths in which arc (i, j) is used. Then the shortest path problem is formulated as follows:

$$\begin{aligned} \min_{x_{ij}} & \sum_{(i,j)\in A} c_{ij}x_{ij} \\ \text{s.t.} & \sum_{j:(i,j)\in A} x_{ij} - \sum_{k:(k,i)\in A} x_{ki} = \begin{cases} n-1 & \text{for } i = s \\ -1 & \text{for all } i \in N \setminus \{s\} \end{cases} \\ & x_{ij} \ge 0 \quad \text{for all } (i,j) \in A \end{aligned} \tag{2}$$

Note the similarities of the LP formulations of the shortest path problem and the minimum cost flow problem. In the shortest path problem there are no capacity constraints on arcs. Also the supply and demand constraints are special: $b(s) = (n-1)$ and $b(i) = -1$ for $i \in N \setminus \{s\}$. The interpretation is that we have $n-1$ supply of units at the source node s and each node $i \in N \setminus \{s\}$ has a demand of 1 unit. Thus, we want to send one unit from the source node s to each other node at minimum cost. So the shortest path problem can be regarded as a special case of the minimum cost flow problem.

Now consider the maximum flow problem formulation, where x_{ij} is the flow on arc (i, j).

$$\max_{x_{ij}} \quad V$$

$$\text{s.t.} \sum_{j:(i,j)\in A} x_{ij} - \sum_{k:(k,i)\in A} x_{ki} = \begin{cases} V & \text{for } i = s \\ 0 & \text{for all } i \in N \setminus \{s,t\} \\ -V & \text{for } i = t \end{cases} \tag{3}$$

$$0 \leq x_{ij} \leq u_{ij} \quad \text{for all } (i,j) \in A$$

Suppose we add an arc (t, s) with infinite capacity $u_{ts} = \infty$, and we let V represent the flow on (t, s): $x_{ts} = V$. Now the objective can be written as $\min -V$ or equivalently $\min -x_{ts}$. Then the maximum flow can be written as follows:

$$\min_{x_{ij}} \quad -x_{ts}$$

$$\text{s.t.} \sum_{j:(i,j)\in A} x_{ij} - \sum_{k:(k,i)\in A} x_{ki} = 0 \quad \text{for all } i \in N \tag{4}$$

$$0 \leq x_{ij} \leq u_{ij} \quad \text{for all } (i,j) \in A$$

The above is clearly a minimum cost flow problem with zero demands and supplies but the zero flow solution is clearly not the optimal solution since one of the arcs (t, s) has a negative cost. Thus we will try and maximize the flow on (t, s) with the result of maximizing the flow between s and t. Note that the above is the circulation problem but with an objective with a negative cost (see lecture notes on maximum flow). Thus, the maximum flow problem is special case of the objective function for the min cost flow problem.

Therefore, the shortest path problem and the maximum flow problem can are both special cases of the minimum cost flow problem. Many algorithms for the minimum cost flow problem take advantage of these connections and use a combination of the special purpose algorithms for the shortest path and the maximum flow problems to solve the minimum cost flow problem. We will not cover these algorithms but the interested reader can check chapter 9 of AMO.

One important link between all three formulations is that each column of the constraint matrix for the conservation of flow equations above, representing the flow on the arc (i, j), has exactly two non-zeroes, represented here as one $+1$ in the row for node i and one -1 in the row for node j.

IV. Characteristics of Solutions

A. *Cycle free Solutions*

We begin with some definitions on our network $G = (N, A)$. Let x be a feasible flow for the minimum cost flow problem as defined in (1). Thus x must satisfy the demand/supply constraints, the non-negativity and capacity constraints. For a feasible flow x some of the arcs will be saturated, some will have zero flow and some will be in between.

Definition An arc (i, j) is a *free arc* if $0 < x_{ij} < u_{ij}$. Otherwise, we call it a *restricted arc*.

So if $x_{ij} = 0$ or $x_{ij} = u_{ij}$, then (i, j) is restricted. When an arc is restricted with $x_{ij} = 0$ then we can only increase the flow; when it is restricted with $x_{ij} = u_{ij}$ then we can only decrease the flow; when an arc is free then we can increase or decrease the flow.

Definition A feasible solution x is a *cycle free solution* if it contains no cycles composed entirely of free arcs (Here we talk about a cycle that ignores the direction of the arcs).

Figure 3 shows a flow on a network that contains a cycle $1-2-3-4-5-1$ composed entirely of free arcs. Thus, the flow shown in Figure 3 is not a cycle free solution.

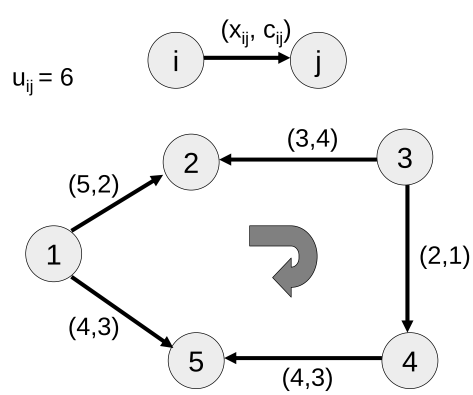

Fig. 3. A feasible flow that is not a cycle free solution: it contains the cycle $1-2-3-4-5-1$ composed entirely of free arcs. Note that the capacity of all the arcs is 6.

It turns out that we can always convert a solution that has a cycle of free arcs to a cycle free solution by adjusting the flows on the cycle without increasing the cost.

We will show this by using the example in Figure 3. We want to change the flow on the cycle arcs such that one of the arcs becomes restricted. This will break the cycle of free arcs. However, we would like to do this without increasing the cost.

Suppose that we increase the flow on the cycle arcs by an amount θ in the clockwise direction as shown in Figure 3. This will increase the flow by θ on the arcs of the cycle that have the same clockwise direction (let us call them *forward arcs*), which are arcs $(1, 2)$, $(3, 4)$, $(4, 5)$. Further, this will decrease the flow by θ on the arcs of the cycle that have the opposite anticlockwise direction (let us call them *backward arcs*), which are arcs $(3, 2)$, $(1, 5)$. The resulting flow after an increase of θ in the clockwise direction is shown in Figure 4.

This increase will still satisfy the conservation of flow constraints on each node since the amount of increase in the outflow will equal to the amount of increase in the inflow at each node. Given that the increase in flow on the cycle does not violate the capacity and non-negativity constraints, the flow remains feasible.

Note that the direction that we have chosen is arbitrary. Since all the arcs on the cycle are free arcs by assumption, we can increase the flow in either direction.

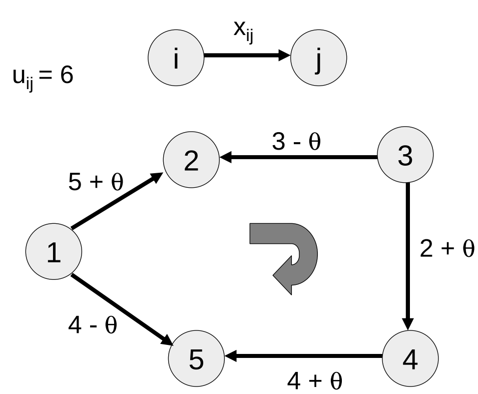

Fig. 4. The resulting flow after increasing the flow on the cycle arcs by θ in the clockwise direction.

Now let us evaluate the cost of this change in flow. We denote the total increase in cost resulting from this change in flow by $\Delta cost$. The increase in cost by increasing the flow on forward arc $(1, 2)$ by an amount θ is $c_{12}\theta = 2\theta$. The decrease in cost by decreasing the flow on backward arc $(3, 2)$ by an amount θ is $c_{32}\theta = 4\theta$. Performing similar calculations on all the cycle arcs we get:

$$\Delta cost = 2\theta + 1\theta + 3\theta - 4\theta - 3\theta = (2 + 1 + 3 - 4 - 3)\theta = -\theta.$$

This means that by increasing the flow by θ in the clockwise direction we increase the cost by $-\theta$ or equivalently that we decrease the cost by θ. In this example for $\theta > 0$, we have $\Delta cost < 0$. Since we would like to minimize cost or equivalently minimize $\Delta cost$, we increase the value of θ as much as we can. The value of θ is bounded from above by the capacities of the forward arcs: $(1, 2)$, $(3, 4)$, $(4, 5)$. It is also bounded from above by the constraint of non-negativity of the backward arcs: $(3, 2)$, $(1, 5)$. This gives us the following constraints on θ:

$$\begin{aligned} \text{capacity of forward arcs} \quad & 5 + \theta \le 6 \quad \text{for arc } (1, 2) \\ & 2 + \theta \le 6 \quad \text{for arc } (3, 4) \\ & 4 + \theta \le 6 \quad \text{for arc } (4, 5) \\ \text{non-negativity of backward arcs} \quad & 3 - \theta \ge 0 \quad \text{for arc } (3, 2) \\ & 4 - \theta \ge 0 \quad \text{for arc } (1, 5) \end{aligned} \tag{5}$$

The lowest upper bound on θ is $\theta \le 1$ and this comes from the forward arc $(1, 2)$. If we set $\theta = 1$ to minimize $\Delta cost$, then arc $(1, 2)$ becomes saturated and thus restricted. Thus we have broken the cycle of free arcs and have reduced the cost by 1 ($\Delta cost = -1$).

In the above example only one arc became restricted in our effort to minimize $\Delta cost$. Sometimes more than one arcs will be become restricted which also breaks the cycle of free arcs.

Further, in the above example minimizing $\Delta cost$ was equivalent to maximizing θ, since the coefficient of θ in $\Delta cost$ was negative. What if this coefficient was positive? For example, suppose that $\Delta cost = 2\theta$. Then minimizing $\Delta cost$ would be equivalent to minimizing θ. Allowing θ to take negative values would be the equivalent of increasing the flow in the opposite direction. The initial direction that we have picked was arbitrary. It might turn out to be the case that the direction that minimizes cost is the opposite direction. In that case we minimize θ and allow θ to take negative value. Then we look for lower bounds on θ which will come from the non-negativity constraints of the forward arcs and the capacity constraints of the backward arcs.

Finally, suppose that $\Delta cost = 0 \times \theta = 0$. Then changing the value of θ makes no difference in the cost. Thus we can change θ to break the cycle without increasing the cost.

In the above method of eliminating cycles of free arcs we have learned that:

1) Since the cycle consists of free arcs we can increase flow in either direction of the cycle and still retain a feasible solution.
2) Increasing flow in one of the directions of the cycle will decrease the cost and the other direction will increase the cost, since $\Delta cost$ and θ have a linear relationship with a positive/negative coefficient. In the case of $\Delta cost = 0$ changing the value of θ in either direction will break the cycle of free arcs but will not alter the cost.
3) By minimizing or maximizing the value of θ according to the direction of reducing cost, we will always hit the value of a capacity or non-negativity constraint of an arc and thus make that arc a restricted arc. This will always result in breaking the cycle of free arcs and it will never increase the cost.

This gives us the following lemma:

Lemma 4.1: In a minimum cost flow problem with bounded optimal solutions, there exist feasible cycle free solutions.

Proof: Suppose there are no cycle free solutions. Take any feasible solution and break each of the cycles with the method described above. The result is a feasible solution with no cycles, i.e. a cycle free solution. This contradicts the initial assumption. ∎

We can also conclude the following:

Theorem 4.2: In a minimum cost flow problem with bounded optimal solutions, there exist an optimal cycle free solution.

Proof: Let x be an optimal solution. Then if x is not cycle free it will contain cycles of free arcs. Break these cycles with the method described above. This will give us a feasible cycle free solution x' without increasing the cost. Thus x' must be optimal. (Note that in this case the $\Delta cost$ of the cycles of free arcs must be 0 since otherwise we would be able to decrease the cost further). ∎

B. Spanning Tree Solutions

We begin with some definitions.

Definition The pair (x, T), where x is a feasible solution and T is a spanning tree on a network G, is called a *spanning tree solution* if every non-tree arc is a restricted arc with respect to flow x.

Note that in a spanning tree solution the tree arcs can be restricted or free arcs.

Definition We call a spanning tree solution (x, T) *nondegenerate* if all the arcs on T are free arcs, and *degenerate* if T contains restricted arcs.

We have the following result:

Lemma 4.3: Given a cycle free solution x we can always convert it to a spanning tree solution (x, T).

Proof: We need to construct the spanning tree T such that every non-tree arc is a restricted arc. Let T be the free arcs of G associated with cycle free solution x. Since x is cycle free then T is a forest. If T is connected then it is a spanning tree and the pair (x, T) is a spanning tree solution. If T is not connected then we add restricted arcs to T that do not form cycles until T is connected and thus a spanning tree. ■

Note that if the forest T of free arcs associated with cycle free flow x, is a spanning tree then only one spanning tree solution is constructed i.e. (x, T). However, if the forest T of free arcs is not connected then there could be many different spanning trees that we can construct and thus many spanning tree solutions. An example of this is shown in Figure 5. Figure 5(a) shows a cycle free solution. Figure 5(b) shows the forest of free arcs. Since the forest is disconnected we can add any restricted arc as long as it does not create a cycle. Figures 5(c) and 5(d) show two different spanning tree solutions that resulted from adding different restricted arcs.

We can conclude the following:

Theorem 4.4: In a minimum cost flow problem with bounded optimal solutions, there exist an optimal spanning tree solution.

Proof: This follows from the fact that there exists an optimal cycle free solution by theorem 4.2, and we can always convert a cycle free solution to a spanning tree solution by lemma 4.3. ■

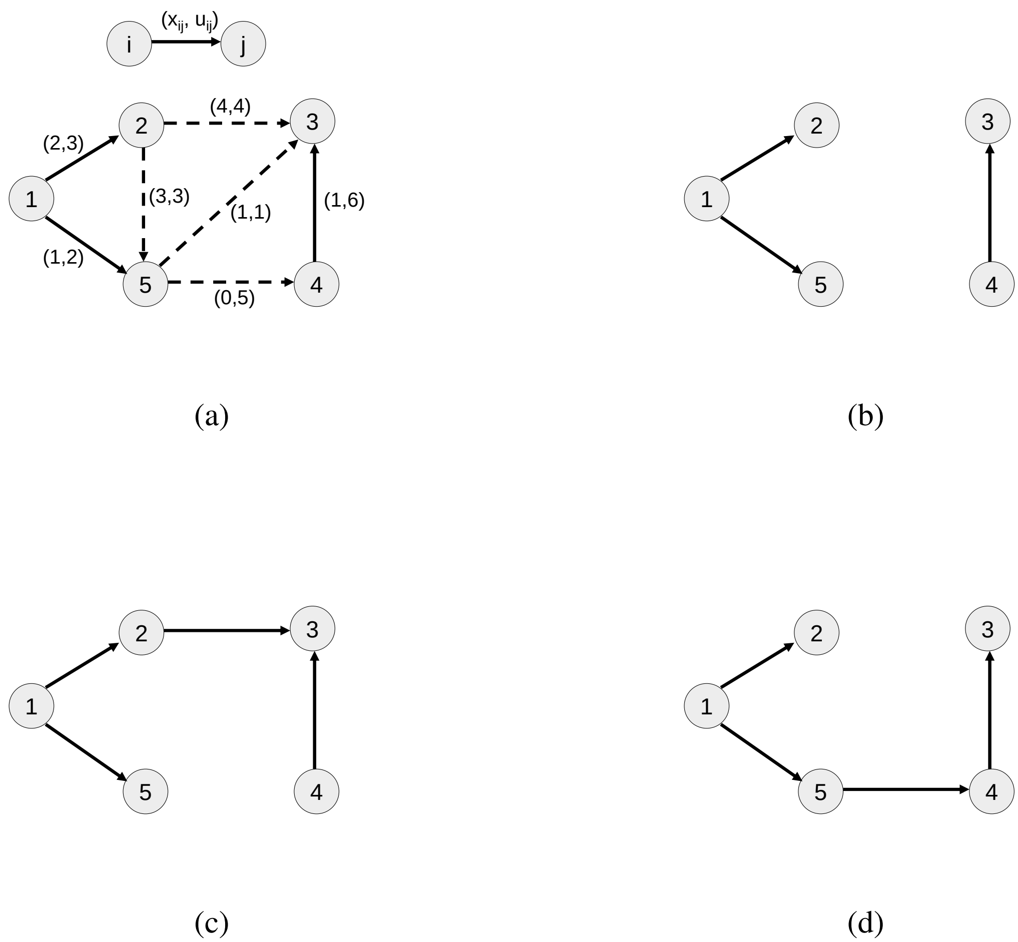

Fig. 5. (a) A cycle free solution; (b) The forest of free arcs; (c) and (d) Two different spanning tree solutions that result from adding different restricted arcs.

A spanning tree solution (x, T) partitions the set of arcs A into three subsets: the set of arcs in T, the non-tree arcs that have $x_{ij} = 0$ which we call L (i.e. at their lower bound), and the non-tree arcs that have $x_{ij} = u_{ij}$ which we call U (i.e. at their upper bound).

Definition We call a partition (T, L, U) of the arcs in $G = (N, A)$, where T is a spanning tree of G, a *spanning tree structure*.

It is easy to see that every spanning tree solution (x, T) defines a unique spanning tree structure (T, L, U).

It turns out that every spanning tree structure (T, L, U) defines a unique flow x and when this flow is feasible, (x, T) is a spanning tree solution. Starting from a spanning tree structure (T, L, U) we will construct a flow x. We first set $x_{ij} = 0$ for all $(i, j) \in L$, and $x_{ij} = u_{ij}$ for all $(i, j) \in U$. There are $n - 1$ arcs remaining that are in T. To find the flow on these arcs we use the conservation of flow constraints on each node:

$$\sum_{j:(i,j)\in A} x_{ij} - \sum_{k:(k,i)\in A} x_{ki} = b(i) \quad \text{for } i \in N \tag{6}$$

If we substitute the values of x_{ij} for $(i, j) \in L, U$ in equations (6), then we are left with $n - 1$ variables (the arcs of T) and n equations (one for each node). Consider the conservation of flow equation for node i that is a leaf node of T, and suppose that i is connected to the remaining tree with arc (i, j). Since at node i the only tree arc is (i, j), the only variable remaining is x_{ij} and thus we can find the value of x_{ij}. We substitute this value in the equations (6) and continue with another leaf node of the remaining tree $T \setminus \{(i, j)\}$. Given that demand must equal supply ($\sum_{i\in N} b(i) = 0$), when we reach the final node of the tree the conservation of flow equation for that node should have been already satisfied.

Thus we have found a flow x that is feasible with respect to equations (6) and every non-tree arc is restricted. However, this flow might not satisfy the constraints $0 \leq x_{ij} \leq u_{ij}$. If it does then the flow x is feasible, and thus (x, T) is a spanning tree solution.

Definition We call a spanning tree structure *feasible* if its unique associated flow (as found above) is feasible.

C. Optimality Conditions

We first start with some definitions. So far we have only considered costs c_{ij} associated with each unit of flow on arc (i, j). Suppose we associate with each node $i \in N$ a cost $\pi(i)$ that we call *node potential* of node i. We denote by $\pi = (\pi(1), \pi(2), \ldots, \pi(n))$ the vector of all node potentials. The interpretation of these node potentials will become clear later. The node potentials $\pi(i)$ will aid us in proving sufficient conditions for optimality of a solution x and also aid us in developing an algorithm for solving the minimum cost flow problem.

Definition Given costs c_{ij} and node potentials $\pi(i)$, we define the *reduced cost* c^{π}_{ij} of arc (i, j)

as follows:

$$c_{ij}^{\pi} = c_{ij} - \pi(i) + \pi(j) \tag{7}$$

We will show next that changing the arc costs to the reduced arc costs does not change the optimal solutions of the minimum cost flow problem. We let,

$$z(\pi) = \sum_{(i,j)\in A} c_{ij}^{\pi} x_{ij}$$

be the value of the optimal solution of the minimum cost flow problem where the arc costs are given by c_{ij}^{π}. Then $z(0)$, with zero node potentials, is the value of the minimum cost flow problem with arc costs c_{ij}.

Lemma 4.5: For any node potentials π, the minimum cost flow problem with arc costs c_{ij} and the minimum cost flow problem with arc costs c_{ij}^{π} have the same optimal solutions. Further,

$$z(\pi) = z(0) - \sum_{k\in N} \pi(k)b(k) \tag{8}$$

Proof: If we show equation (8), then we know that minimizing $z(\pi)$ is the same as minimizing $z(0)$ since they only differ by a constant. We have,

$$z(\pi) = \sum_{(i,j)\in A} c_{ij}^{\pi} x_{ij} = \sum_{(i,j)\in A} \left(c_{ij} - \pi(i) + \pi(j)\right) x_{ij} = z(0) - \sum_{(i,j)\in A} \pi(i)x_{ij} + \sum_{(i,j)\in A} \pi(j)x_{ij}$$

Now consider the last two summations: for each node $k \in N$, the only terms that have the factor $\pi(k)$ are only associated with arcs that either come into k or leave node k. The terms that have the term $\pi(k)$ are as follows:

$$\sum_{i:(i,k)\in A} \pi(k)x_{ik} - \sum_{j:(k,j)\in A} \pi(k)x_{kj} = -\pi(k)b(k)$$

From the conservation of flow equations we can see that these terms are the inflow minus the outflow which is equals to $-b(k)$. Performing this for all nodes we get the desired result. ■

Thus we have seen that we can introduce any node potentials π and change the costs to reduced costs and still retain the same optimal solution.

Now we are ready to introduce sufficient conditions for a spanning tree solution (x, T) or the corresponding spanning tree structure (T, L, U) to be optimal.

Theorem 4.6: Optimality conditions A spanning tree structure (T, L, U), and its associated spanning tree solution (x^*, T), solves the minimum cost flow problem optimally if it is feasible and for some choice of node potentials π, its reduced cost satisfies the following three conditions:

$$c_{ij}^{\pi} = c_{ij} - \pi(i) + \pi(j) = 0 \quad \text{for any } (i,j) \in T \tag{9}$$

$$c_{ij}^{\pi} = c_{ij} - \pi(i) + \pi(j) \geq 0 \quad \text{for any } (i,j) \in L \tag{10}$$

$$c_{ij}^{\pi} = c_{ij} - \pi(i) + \pi(j) \leq 0 \quad \text{for any } (i,j) \in U \tag{11}$$

Proof: Suppose the spanning tree structure (T, L, U) is feasible and the associated spanning tree solution (x^*, T) satisfies conditions (9)-(11) above for some node potential π. We will show that x^* is optimal. In lemma 4.5 we showed that minimizing $\sum_{(i,j)\in A} c_{ij}x_{ij}$ is the same as minimizing $\sum_{(i,j)\in A} c_{ij}^{\pi}x_{ij}$ for any π. Thus we will show that x^* minimizes the objective $\sum_{(i,j)\in A} c_{ij}^{\pi}x_{ij}$ for the specific π given above. Substituting c^{π} from conditions (9)-(11) gives the objective of:

$$\text{minimize} \sum_{(i,j)\in L} c_{ij}^{\pi}x_{ij} - \sum_{(i,j)\in U} |c_{ij}^{\pi}|x_{ij}$$

Now consider any arbitrary flow x. Since x^* by definition has $x_{ij}^* = 0 \leq x_{ij}$ for $(i,j) \in L$ and $x_{ij}^* = u_{ij} \geq x_{ij}$ for $(i,j) \in U$, we must have that x^* has smaller cost than x. Thus, x^* is optimal. ∎

We have shown that there exists an optimal spanning tree solution. Further, we have derived optimality conditions for a spanning tree solution. In the next section we will devise an algorithm that moves from one spanning tree solution to the next checking the optimality conditions for each solution.

V. Network Simplex Algorithm

In this section we develop the *network simplex algorithm* based on concepts that we have developed in the previous section. Given that there exists an optimal spanning tree solution, we will start from an initial spanning tree solution and check its optimality using some node potentials π and the resulting reduced costs c^{π}. We can always find node potentials π such that our current spanning tree solution (x, T) satisfies the first optimality condition: $c_{ij}^{\pi} = c_{ij} - \pi(i) + \pi(j) = 0$ or equivalently $\pi(i) = \pi(j) + c_{ij}$ for $(i,j) \in T$. Then we will check whether the non-tree arcs in L and U satisfy the second and third optimality conditions. If they do not then we add a violating non-tree arc into the tree T and remove a tree arc, thus changing the spanning tree structure (T, L, U). We repeat again with the new spanning tree solution.

A. *Finding an initial spanning tree solution*

We will describe two ways to find an initial spanning tree solution.

The first is based on assumption 6) that we stated in section II-B. We assumed that we add two artificial arcs with sufficiently large costs and capacities between node 1 and every other node $j \neq 1$: $(1, j)$ and $(j, 1)$. In this case the initial spanning tree solution can be constructed as follows: for $b(j) \geq 0$ include arc $(j, 1)$ in T and set $x_{j1} = b(j)$, and for $b(j) < 0$ include arc $(1, j)$ in T and set $x_{1j} = -b(j)$. Set all other $x_{ij} = 0$. This solution satisfies demand/supply and capacity constraints and it is a spanning tree solution.

The second is based on assumption 4) of section II-B. There we describe a way to find a feasible solution to the minimum cost flow problem by converting it into a maximum flow problem. If we find a general feasible solution, then we have shown in section IV-A how to convert it to a cycle free solution and then in section IV-B how to convert that into a spanning tree solution.

B. *Checking optimality of spanning tree solutions*

Given a spanning tree solution (x, T) with a spanning tree structure (T, L, U), we can always find node potentials π to satisfy the first optimality condition:

$$c_{ij} - \pi(i) + \pi(j) = 0 \quad => \quad \pi(i) = \pi(j) + c_{ij} \quad (i, j) \in T.$$

We would like to satisfy the above $n - 1$ equations (one for each arc of the tree T) and we have n variables, the node potentials $\pi(i)$ for $i \in N$. There are infinitely many solutions to this system of equation but note that adding a constant K to all node potentials cancels out and still gives a solution to the system of equations. Thus only the relative values of the node potentials matter. Thus we can safely fix: $\pi(1) = 0$. Given that we know the node potential $\pi(i)$ of node i, we can use $\pi(i) = \pi(j) + c_{ij}$ on each tree arc (i, j) to find the node potential of node j; if the tree arc is (j, i), the we use $\pi(j) = \pi(i) + c_{ji}$ to find $\pi(j)$. This guarantees that we satisfy the first optimality conditions.

An example of this is shown in Figure 6. We start with $\pi(1) = 0$. For tree arc $(1, 2)$, we have $\pi(2) = \pi(1) - c_{12} = -5$. Similarly, for tree arc $(1, 3)$ we have $\pi(3) = \pi(1) - c_{13} = -3$. Now for tree arc $(4, 2)$, we use $\pi(4) = \pi(2) + c_{42} = -5 + 1 = -4$. The rule of thumb is that we subtract the cost when we arc is forward and add the cost when the arc is backward.

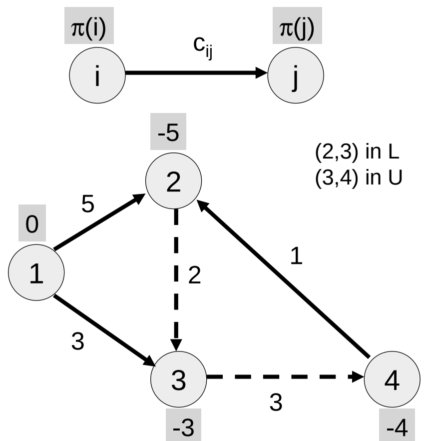

Fig. 6. Using $\pi(1) = 0$ and $\pi(i) = \pi(j) + c_{ij}$ to find the node potentials on the network where the arcs in T are shown in solid lines.

Let us try and interpret the above. We found the node potentials π by starting with $\pi(1) = 0$. All the other node potentials will be relative to $\pi(1)$. Then we calculated $\pi(2) = \pi(1) - c_{12} = -5$. Since $c_{ij} = 5$ is the cost of sending one unit of flow from node 1 to node 2, $-c_{ij} = -5$ is the cost of sending one unit of flow from 2 to 1 (i.e. reducing flow on arc $(1, 2)$). Now consider node 4:

$$\pi(4) = \pi(2) + c_{42} = \pi(1) - c_{12} + c_{42} = -5 + 1$$

The node potential for node 4 is calculated by taking the unique path on T from node 4 to node 1 and calculating the cost of sending one unit of flow from node 4 to node 1. For all forward arcs on the path from 4 to 1 the cost is added and for all backward arcs the cost is subtracted. Thus, the interpretation of $\pi(i)$ is the cost of sending a unit of flow from i to 1 on the unique path on T.

The difference between two node potentials also has a good interpretation:

$$\pi(4) - \pi(3) = (\pi(1) - c_{12} + c_{42}) - (\pi(1) - c_{13}) = c_{42} - c12 + c_{13}$$

The above is the cost of sending one unit of flow from node 4 to node 3 by using their unique path in T. Again we add the costs of forward arcs of the path and subtract the cost of backward arcs on the path.

Now we are ready to check whether the remaining two conditions are satisfied:

$$c_{ij} \geq \pi(i) - \pi(j) \quad \text{for any } (i,j) \in L$$

$$c_{ij} \leq \pi(i) - \pi(j) \quad \text{for any } (i,j) \in U$$

Let us try and interpret these conditions. The first one states that for a non-tree arc (i,j) with zero flow, the cost of sending a unit of flow through arc (i,j) is greater than sending the flow from i to j through the unique path in T. So sending flow directly is more expensive than using the tree T. Since this is a relatively expensive arc it makes sense to keep its flow to zero i.e. keep it in L. The second condition states that for a non-tree saturated arc (i,j), the cost of sending a unit of flow through arc (i,j) is less than sending the flow from i to j through the unique path in T. So sending flow directly is cheaper than using the tree T. Since this is an relatively cheap arc it makes sense to keep its flow at capacity i.e. keep it in U.

If the non-tree arcs satisfy these conditions with the node potentials that we derived, then the current solution is optimal. If some arcs do not satisfy these conditions then we need to improve our spanning tree solution.

C. Updating the Spanning Tree Solution

Suppose that a set of non-tree arcs have not satisfied the optimality conditions. For an arc $(i,j) \in L$, this means that it is cheaper for the flow to pass through arc (i,j) than through the tree T. The difference in costs per unit flow is given by $\pi(i) - \pi(j) - c_{ij} = -c^{\pi}_{ij} \geq 0$. Further, for an arc $(i,j) \in U$, this means that it is cheaper for the flow to pass through the tree T than through arc (i,j). The difference in costs per unit flow is give by $c_{ij} - \pi(i) + \pi(j) = c^{\pi}_{ij} \geq 0$. Thus, for each violating arc (i,j) the absolute reduced cost $|c^{\pi}_{ij}|$ gives us the cost reduction per unit flow of an increase in flow (of a zero flow arc) or a decrease in flow (of a saturated arc). It would be beneficial if we added these arcs to our tree T and converted them into free arcs.

There are many ways of choosing a violated arc to add to the tree T. We will use Dantzig's pivot rule that chooses the arc with the maximum absolute reduced cost (maximum $|c^{\pi}_{ij}|$). This ensures that our new solution will have the maximum improvement in cost but has the drawback that we need to check all arcs for violations. Another faster rule is to just enter the first violated arc that we find. Other pivot rules can be found in AMO, chapter 11.

Suppose we picked violated non-tree arc (i, j) and we add it to the tree T. This will create a cycle which we call the *pivot cycle*. If $(i, j) \in L$, then we increase the flow of the cycle as much as possible in the direction that will increase the flow of arc (i, j). If $(i, j) \in U$, then we increase the flow of the cycle as much as possible in the direction that will decrease the flow of arc (i, j). Suppose that we increase the flow on the pivot cycle by θ.

Three things can happen:

1) A tree arc (k, l) either gets saturated or gets zero flow. In this case we move (k, l) to L or U accordingly and add arc (i, j) to the tree T. If there are more than one such arcs then we pick one of them according to a tie-breaking rule (we can give priority to the one with the lowest index of its head node and if that is the same, then the lowest index of its tail node). When there are more than one such arcs, the new spanning tree solution will be degenerate.
2) Arc (i, j) gets saturated at the opposite end. If (i, j) was in L we now put it in U and vice versa. In this case we keep the same spanning tree T, but the flows on T have changed along with the flow on (i, j). Even though the node potentials in the next iteration will remain the same, the sets L, U have changed.
3) We cannot increase the flow on the pivot cycle because one of the tree arcs in T was restricted i.e. $\theta = 0$. This is called a *degenerate iteration* and it can only occur if the spanning tree T is a degenerate spanning tree. In this case, (i, j) enters the tree T and we remove the other restricted arc of the tree and enter it in L or U accordingly.

If we increase the flow on the pivot cycle by θ then our decrease in cost will be $\theta|c^{\pi}_{ij}|$.

With the new spanning tree solution (x, T) with the updated flow x and tree arcs T, we can again calculate node potentials and check the optimality conditions.

D. *The Algorithm*

The details of the algorithm have been explained in the previous sections. A general description of the pseudo code is shown in Algorithm 1.

If all iterations of the algorithm are nondegenerate then the algorithm terminates in a finite number of steps. However, degeneracy poses a problem. There is a way to overcome this using

Algorithm 1 Network Simplex Algorithm

begin

  Find an initial feasible spanning tree structure (T, L, U) with spanning tree solution (x, T).

  Let π be the node potentials associated with this spanning tree structure.

  while some non-tree arc violates the optimality conditions

    Select an entering arc (k, l) that violates its optimality conditions

    Add arc (k, l) to the tree T and determine a leaving arc (p, q)

    Update the tree structure (T, L, U), the flow x and the node potentials π

  end;

end;

strongly feasible spanning trees. We will not cover this in this lecture but for the interested reader check section 11.6 of AMO.

The network simplex algorithm is a special case of the simplex algorithm for linear programming. However, the special structure of the network allows us to solve it more efficiently than if we used the generic simplex method.

E. An example

We illustrate the network simplex algorithm on the network shown in Figure 7.

First we find an initial spanning tree solution. To do that we have solved a maximum flow problem from source node 1 to sink node 6. Here it was not necessary to add dummy source and sink nodes since the only supply is at node 1 and the only demand is at node 6. The solution x of the maximum flow problem is shown in Figure 8(a). This solution has resulted in free and restricted arcs. Since the free arcs do not form a cycle this is a cycle free solution. To convert it into a spanning tree solution we look at the free arcs: $(1, 2)$, $(2, 4)$, $(2, 5)$, $(5, 6)$. Here the free arcs are not connected and thus we add arc $(1, 3)$ to T to make it a spanning tree. Then (x, T) is a spanning tree solution shown in Figure 8(b).

Next we calculate the node potentials on the tree T by setting $\pi(1) = 0$ and traversing through the tree T arcs using $\pi(i) = \pi(j) + c_{ij}$. The resulting node potentials are shown in Figure 8(b).

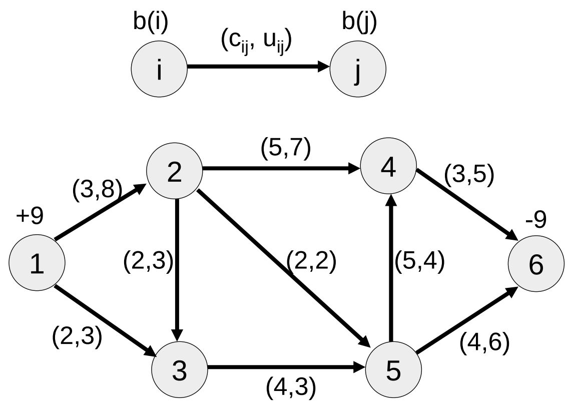

Fig. 7. A network with costs and capacities shown on each arc and demand/supplies shown on each node.

Now we check the optimality conditions for the non-tree arcs:

$$
\begin{aligned}
\text{Arcs in } L\text{:} \quad & c_{23} = 2 \geq \pi(2) - \pi(3) = -3 - (-2) = -1 \\
& c_{54} = 5 \geq \pi(5) - \pi(4) = -5 - (-8) = 3 \\
\text{Arcs in } U\text{:} \quad & c_{35} = 4 \leq \pi(3) - \pi(5) = -2 - (-5) = 3 \\
& c_{46} = 3 \leq \pi(4) - \pi(6) = -8 - (-9) = 1
\end{aligned}
\tag{12}
$$

As we can see both arcs in U are violated and the most violated is arc $(4, 6)$ with $|c^{\pi}_{46}| = 2$. If we were using Dantzig's pivot rule then we would enter this arc. However, following the example of AMO, we enter arc $(3, 5)$ in T, which is the first arc that we found that was violated. Since arc $(3, 5) \in U$ is at its upper bound we want to reduce its flow. The pivot cycle is shown in figure 8(c) and the direction shown is the direction that is opposite to the arc $(3, 5)$ since we want to reduce its flow. We increase the flow by θ in the clockwise direction. The bounds on theta as as follows:

$$
\begin{aligned}
\text{for arc } (1, 2)\text{:} \quad & 6 + \theta \leq 8 \\
\text{for arc } (2, 5)\text{:} \quad & 1 + \theta \leq 2 \\
\text{for arc } (3, 5)\text{:} \quad & 3 - \theta \geq 0 \\
\text{for arc } (1, 3)\text{:} \quad & 3 - \theta \geq 0
\end{aligned}
$$

The lowest upper bound on θ is given by $\theta \leq 1$ from arc $(2,5)$. Thus we set $\theta = 1$ and note that arc $(2,5)$ is now saturated and thus it leaves T and enters U. We update the flow x and the spanning tree structure (T, L, U) and this is shown in Figure 8(d). We calculate the node potentials associated with the new T. These are shown in Figure 8(d). Now we check the optimality conditions for the non-tree arcs:

$$\begin{aligned}
\text{Arcs in } L\text{:} \quad & c_{23} = 2 \geq \pi(2) - \pi(3) = -3 - (-2) = -1 \\
& c_{54} = 5 \geq \pi(5) - \pi(4) = -6 - (-8) = 2 \\
\text{Arcs in } U\text{:} \quad & c_{25} = 2 \leq \pi(2) - \pi(5) = -3 - (-6) = 3 \\
& c_{46} = 3 \leq \pi(4) - \pi(6) = -8 - (-10) = 2
\end{aligned} \tag{13}$$

The only violation occurs with arc $(4,6) \in U$ with $|c^{\pi}_{46}| = 1$ (note that this has changed since the last iteration). We enter arc $(4,6)$ into T and create the pivot cycle which is shown in Figure 8(e). We would like to decrease the flow of $(4,6)$ and thus we pick the direction of the pivot cycle to be opposite to the direction of $(4,6)$. The direction is shown in Figure 8(e). As shown we increase the flow in this direction by θ. We would like to find the maximum value of θ. The upper bounds on theta are as follows:

$$\begin{aligned}
\text{for arc } (1,2)\text{:} \quad & 7 - \theta \geq 0 \\
\text{for arc } (2,4)\text{:} \quad & 5 - \theta \geq 0 \\
\text{for arc } (4,6)\text{:} \quad & 5 - \theta \geq 0 \\
\text{for arc } (1,3)\text{:} \quad & 2 + \theta \leq 3 \\
\text{for arc } (3,5)\text{:} \quad & 2 + \theta \leq 3 \\
\text{for arc } (5,6)\text{:} \quad & 4 + \theta \leq 6
\end{aligned}$$

The lowest upper bound on θ is given by $\theta \leq 1$ from arcs $(1,3)$ and $(3,5)$. Thus we set $\theta = 1$ and we choose to remove arc $(3,5)$ by putting it into U. We update the flow x and the spanning tree structure and calculate the new node potentials, all of which are shown in Figure 8(f). Now we check the optimality conditions for non-tree arcs:

$$\begin{aligned} \text{Arcs in } L\text{:}\quad & c_{23} = 2 \geq \pi(2) - \pi(3) = -3 - (-2) = -1 \\ & c_{54} = 5 \geq \pi(5) - \pi(4) = -7 - (-8) = 1 \\ \text{Arcs in } U\text{:}\quad & c_{25} = 2 \leq \pi(2) - \pi(5) = -3 - (-7) = 4 \\ & c_{35} = 4 \leq \pi(3) - \pi(5) = -2 - (-7) = 5 \end{aligned} \tag{14}$$

Since all optimality conditions are satisfied the current spanning tree solution (x, T) is optimal.

VI. Integrality Theorems and Duality

From the algorithm above we note that if the capacities and demand/supplies are integer, then our initial solution x will be integer since we solve a maximum flow problem with integer capacities (where the demands/supplies are used as capacities on dummy arcs). If we need to break any cycles then we increase the flow on a cycle by an integer amount that is determined by the integer capacities. Thus the initial spanning tree solution (x, T) is integer. Then the change in flow occurs when we add a restricted arc to the tree T and increase the flow as much as possible in a direction of the pivot cycle. Since again this increase in flow is bounded by the capacities of arcs, it will be integer. Thus, the resulting flow x will remain integer. This gives us the following result:

Theorem 6.1: Primal Integrality Theorem If the capacities and the demand/supplies in a minimum cost flow problem are integer, then there exists an integer optimal flow.

We can also claim the same about the node potentials π. If the arc costs are integer, then the node potentials are integer.

Theorem 6.2: Dual Integrality Theorem If the costs in a minimum cost flow problem are integer, then there exist optimal integer node potentials.

The above is called the dual integrality theorem because the node potentials π are in fact the dual variables of the minimum cost flow problem. I ask you to think about the duality of the minimum cost flow problem in exercise 1.

VII. Applications

For numerous applications you can check section 9.2 of AMO. Sections 11.7 and 11.8 apply the network simplex algorithm to the shortest path problem and the maximum flow problem.

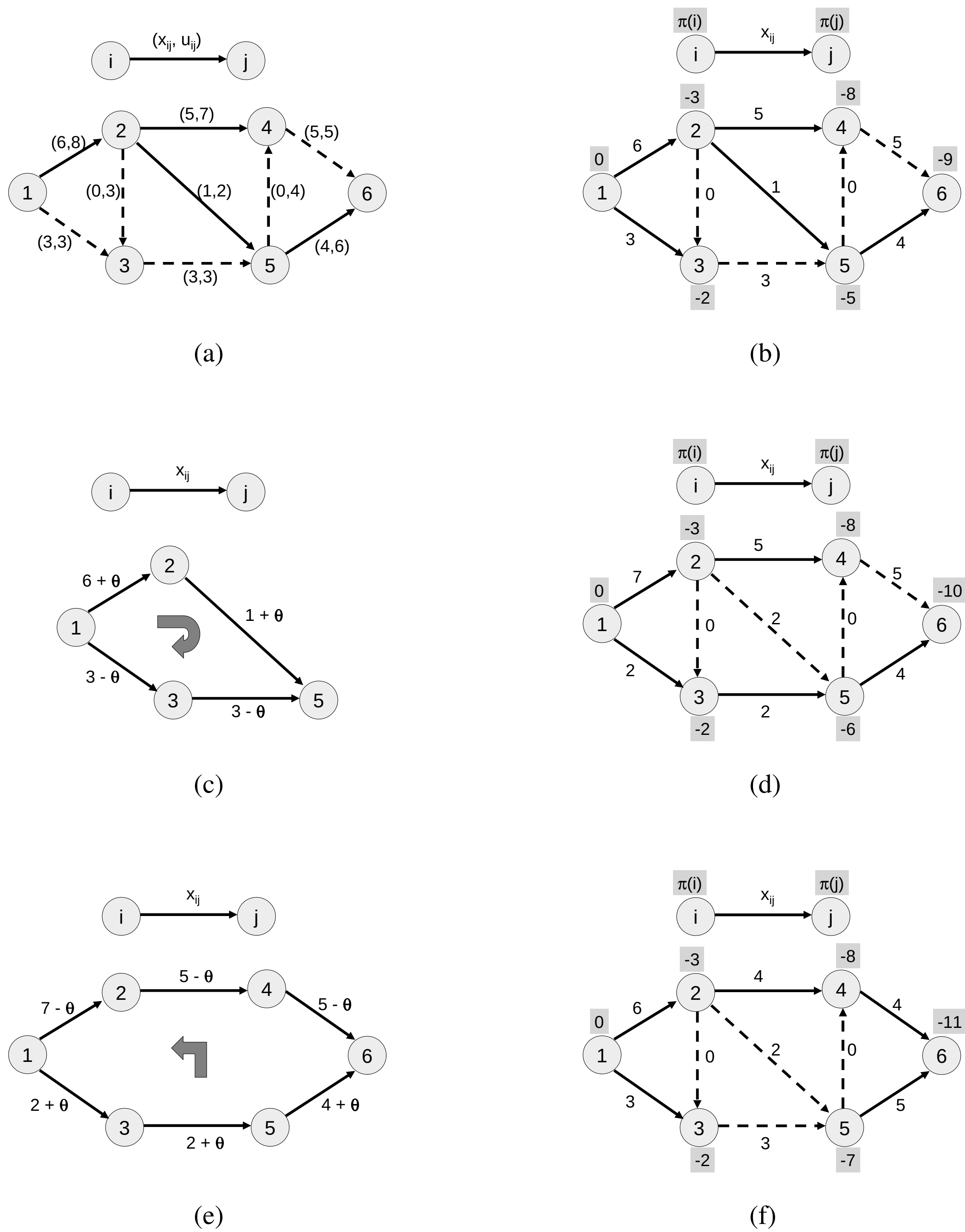

Fig. 8. (a) The free arcs do not form a spanning tree and thus we add arc $(1,3)$ to T. (b) Compute the node potentials and check optimality conditions. (c) Arc $(3,5)$ violates the optimality conditions and thus enters the tree; we increase the flow in the direction shown with maximum possible increase $\theta = 1$. (d) Arc $(2,5)$ leaves the tree T, and the new spanning tree solution (x,T) is shown along with the new node potentials. (e) Arc $(4,6)$ violates the optimality conditions and thus enters the tree; we increase the flow in the direction shown with maximum possible increase $\theta = 1$. (f) Arc $(3,5)$ leaves the tree T, and the new spanning tree solution (x,T) is shown along with the new node potentials. This solution is optimal.

VIII. Exercises

1) Give an economic interpretation of the optimality conditions for non-tree arcs. Find the dual of the min cost flow problem and use complementary slackness to derive the optimality conditions for a non-degenerate spanning tree structure.
2) Solve 11.2 of AMO.
3) Solve 11.12 of AMO.
4) Solve 11.16 of AMO.