Chapter 1

Introduction

1.1 Introduction

Networks appear everywhere in our daily lives. Think about electrical and power networks, telephone networks, highway systems, rail networks, airline service networks, manufacturing and distribution networks, computer networks, etc. In these problem domains, we wish to move some entity (electricity, consumer products, people, vehicles) from one point to another in the underlying network and to do so as efficiently as possible, both to provide good service to the users and use the underlying transmission facilities effectively.

In this course, we learn how to model application settings as mathematical objects known as Network Flow Problem (NFP) and study algorithms to solve the resulting models. NFP are studied in several research domains, including applied mathematics, computer science, engineering, management, and operations research. We wish to address basic questions, such as:

1. Shortest Path Problem (SPP): What is the best way to traverse a network to get from one point to another as cheaply as possible?

2. Maximum Flow Problem (MFP): If a network has capacities on arc flows, how can we send as much flow as possible between two points while honoring the arc flow capacities?

3. Minimum Cost Flow Problem (MCFP): If we incur a cost per unit flow on a network with arc capacities and need to send units of a good that reside at one or more points in the network to one or more other points, how can we send the material at minimum possible cost?

From a pure mathematical perspective, most of these problems are trivial to solve as we need to consider a finite number of alternatives. A traditional mathematician might argue that the problems are well solved: simply enumerate all feasible solutions, and choose the best one. Unfortunately, this approach is inapplicable since the number of solutions can be very large. So instead, we devise algorithms that are *good*, i.e., whose computation time is *reasonable* for problems met in practice.

1.2 Network Flow Problems

The Minimum Cost Flow Problem (MCFP) is a fundamental, easy-to-state problem: we wish to find a least-cost shipment of a commodity through a network to satisfy demands at certain nodes from available supplies at other nodes. The MCFP has many applications: the distribution of a product from manufacturing plants to warehouses or from warehouses to retailers; the flow of raw material and intermediate goods through machine stations in a production line; the routing of automobiles through an urban street network; and the routing of calls through the telephone system.

Let $G = (N, A)$ be a directed network defined by a set N of n nodes and a set A of m arcs. Each arc $(i, j) \in A$ has an associated cost c_{ij} that denotes the cost per unit flow on that arc – the flow cost varies linearly with the flow. We also associate, with each arc $(i, j) \in A$, a capacity u_{ij} (i.e., the maximum flow on the arc) and a lower bound l_{ij} (i.e., the minimum flow on the arc). We associate, with each node $i \in N$, an integer number b_i representing its supply/demand. If $b_i > 0$, node i is a supply node; if $b_i < 0$, node i is a demand node with a demand of $-b_i$; and if $b_i = 0$, node i is a transshipment node. We assume $\sum_{i \in N} b_i = 0$. The decision variables in the MCFP are arc flows, represented by flow variables $x_{ij} \in \mathbb{R}_+$, for each arc $(i, j) \in A$. The MCFP can be formulated as follows

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \tag{1.1a}$$

$$\text{s.t.} \sum_{j \in N : (i,j) \in A} x_{ij} - \sum_{j \in N : (j,i) \in A} x_{ji} = b_i \qquad \forall i \in N \tag{1.1b}$$

$$l_{ij} \leq x_{ij} \leq u_{ij} \qquad \forall (i, j) \in A \tag{1.1c}$$

Constraints (1.1b) are called mass balance constraints, stating that, for each node, its outflow (i.e., the flow emanating from it) minus its inflow (i.e., the flow entering the node) must be equal to b_i. Constraints (1.1c) stipulate that the flow of each arc must be between its lower bound and its capacity.

In most problems we consider, we assume that the data (i.e., arc capacities, arc costs, supply, demands, etc) are integral. This is the integrality assumption, which is not restrictive because (i) we can always transform rational data to integer data by multiplying them by a suitably large number, and (ii) we need to convert irrational numbers to rational numbers to represent them on a computer.

Some special cases of the MCFP play a central role in network optimization.

1.2.1 Shortest Path Problem

In the Shortest Path Problem (SPP), we wish to find a min-cost path from a specified source node s to another specified sink node t, assuming that each arc $(i, j) \in A$ has an associated cost c_{ij}. SPP arises in many problem domains, e.g., equipment replacement, project scheduling, cash flow management, message routing, and traffic flow. If we set $b_s = 1$, $b_t = -1$, and $b_i = 0$ for all other nodes in the MCFP, and we set $l_{ij} = 0$ and $u_{ij} = 1$, for each arc $(i, j) \in A$, we get the SPP.

1.2.2 Assignment Problem

The data of the Assignment Problem (AP) consist of two equally sized sets N_1 and N_2 (i.e., $|N_1| = |N_2|$), a collection of pairs $A \subseteq N_1 \times N_2$ representing possible assignments, and a cost c_{ij} associated with each $(i, j) \in A$. In the AP, we wish to pair, at minimum cost, each object in N_1 with exactly one object in N_2, and vice versa. Examples of the AP include assigning people to projects, jobs to machines, tenants to apartments, swimmers to events in a swimming meet, and school graduates to available internships. The AP is a MCFP in a network $G = (N_1 \cup N_2, A)$ with $b_i = 1$ for all $i \in N_1$, $b_i = -1$ for all $i \in N_2$, and $u_{ij} = 1$ for all $(i, j) \in A$.