M.O. Ball et al., Eds., *Handbooks in OR & MS, Vol. 7*

Chapter 2

Primal Simplex Algorithms for Minimum Cost Network Flows

Richard V. Helgason and Jeffery L. Kennington
Department of Computer Science and Engineering, Southern Methodist University, Dallas, Texas 75275, U.S.A.

1. Introduction

Due to the special structure of bases for the linear network flow model, specialized simplex-based software can solve these problems in from one to two orders of magnitude faster than general linear programming software. The objectives of this chapter are to (1) summarize the ideas fundamental to efficient software implementation of the primal simplex algorithm for minimum cost network flow problems and (2) indicate how these ideas have been extended for generalized networks, multicommodity networks, and networks with arbitrary side constraints.

1.1. Set notation

For the most part we adopt standard set notation conventions. Sets will usually be denoted by upper case Roman letters such as X. The empty set will be denoted by Φ. For a finite set X we let $\#X$ denote the number of elements in X. We let $\imath n = \{1, \ldots, n\}$ and $\jmath n = \{0, \ldots, n\}$. Given set X, we define the *equality relation on* X to be $X \doteq X \equiv \{(x, x) : x \in X\}$. We will also use multisets in which a repetition factor is allowed for set elements. For a finite multiset Y then, $\#Y$ will also incorporate multiplicities.

1.2. Matrix and vector notation

Matrices will usually be denoted by upper case Roman letters such as A. Row vectors will usually be denoted by lower case Greek letters such as π. Column vectors will usually be denoted by lower case Roman letters such as x. The element in the ith row and jth column of matrix A will be denoted by A_{ij}. The row (column) vector whose entries are from the ith row (jth column) of A will be denoted by $A_{i.}$ ($A_{.j}$). The ith element of a vector such as x will be denoted by x_i. We will allow extensive subscripting and superscripting of matrices and vectors for identification purposes. Inasmuch as this may interfere with the subscripting convention for element identification above, we also adopt the functional notation $(\cdot)_i$ and $(\cdot)_{ij}$ for vector and matrix element identification, respectively, so that $(X_{ij})_{pq}$ is the pqth element of matrix X_{ij}. We will use e_i (e^i) for the column (row) vector whose ith element is a 1 and whose other elements are all zeros. We will use e_{ij} to denote the column vector whose ith element is a 1, whose jth element is a -1, and whose other elements are all zeros, so that e_{ij} is $e_i - e_j$. We will use $\hat{0}$ and $\hat{1}$ as row or column vectors with orientation and dimension given by context, having as uniform elements 0 or 1, respectively. We abuse notation by allowing $\imath n = \{1, \ldots, n\}$ to also be used as a row vector. The *diagonal* of matrix A is the set of elements $\{A_{ii}\}$. The matrix A is said to be *upper* (*lower*) *triangular* if $A_{ij} = 0$ when $j > i$ $(i > j)$ and, more simply, *triangular* in either case. The matrix A is said to be *diagonal* if it is both upper and lower triangular. A triangular matrix will be nonsingular when its diagonal elements are all nonzero. The matrix A is said to be *triangularizable* if it can be brought to nonsingular triangular form by a sequence of row and column interchanges.

1.3. Graph notation

We define a set of *nodes* or *vertices* V to be any set of consecutive integers which we typically take to be $\imath n$ or $\jmath n$. Given a set of nodes V, we define an *arc* or *edge* for V to be any ordered pair (i, j) with $i \in V$, $j \in V$, and $i \neq j$. The arc (i, j) is said to be *incident on* (touch) both i and j, to *connect* i and j (or j and i), and to be *directed from i to j*. Formally, a *network* or *directed graph* is defined to be $G = \langle V, E \rangle$ where V is a set of nodes and E is a set of arcs for V. Apparently then $E \subseteq (V \times V) \setminus (V \doteq V)$. When $V = \Phi$ then also $E = \Phi$ and in this case G is called the *trivial* graph. We shall also allow E to be a multiset when it is desirable to have more than one arc connect two nodes. In this case one could more properly refer to G as a *multigraph*. For $\#E = m$, we will find it convenient to label the arcs with elements from $\imath m$.

1.4. Visual representation

The nodes of a network may be viewed as locations or terminals that a given commodity can be moved from, to, or through and the arcs of a network may be viewed as unidirectional means of commodity transport connecting or serving those nodes. Hence arcs may represent streets and highways in an urban transportation network, pipes in a water distribution network, or telephone lines in a communication network. The structure of the network can be displayed by means of a labeled drawing in which nodes are represented by circles or boxes and arcs are represented by line segments incident on two nodes. Each line segment will have an arrowhead placed somewhere on it to indicate the direction of the associated commodity transport. Typically the arrowhead will be incident on the node to which the commodity is being transported. An example network illustration is given in Figure 1.