ALGORITHMS FOR MAXIMAL FLOW PROBLEMS

Thesis submitted in partial fulfillment of the requirement for

the award of the degree of

Masters of Science

in

Mathematics and Computing

Submitted by

Ishu Rani

Roll no. - 300903007

Under

the guidance of

Dr. Mahesh Kumar Sharma

JULY 2011

School of Mathematics and Computer Applications

Thapar University

Patiala-147004 (Punjab)

INDIA

DEDICATED

TO

GOD, MY PARENTS

AND

MY SUPERVISOR

CERTIFICATE

I hereby certify that the work presented in the thesis entitled " ALGORITHMS FOR MAXIMAL FLOW PROBLEM " which is being presented for the award of degree of Master of Science, School of Mathematics and Computer Applications, Thapar University, Patiala is an authentic record of my own work carried out under the supervision of Dr. Mahesh Kumar Sharma.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

Ishu Rani

(Ishu Rani)

Reg. No. 300903007

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(Dr. Mahesh Kumar Sharma)
Assistant Professor
SMCA, Thapar University
Patiala.

Countersigned by:

Dr. S.S. Bhatia 15/7
(Professor & Head)
School of Mathematics & Computer Applications
Thapar University, Patiala.

Dr. S.K. Mohapatra
Dean of Academic Affairs
Thapar University
Patiala.

ACKNOWLEDGEMENT

The completion of this thesis has involved a lot of people to whom I would like to express my sincere thanks and gratitude for their help.

For most, I would like to pass my appreciation and gratitude to my honorable supervisor, Dr. Mahesh Kumar Sharma, Assistant Professor, School of Mathematics and Computer Applications, Thapar University, Patiala. For their constructive suggestions, detailed corrections, support and encouragement in accomplishing this work. Moreover, for mentoring me when I needed the most. I am fortunate that I got an opportunity to work under their supervision.

I express my regards and gratitude to Dr. S.S. Bhatia, Head of Department, School of Mathematics and Computer Applications, Thapar University, Patiala, for providing keen interest, unfailing support, inspiration and necessary research facilities in the school.

I am thankful to Mr. Gourav Gupta, School of Mathematics and Computer Applications, Thapar University, Patiala for his help when it is need to complete thesis.

I would like to thank my beloved parents for their unconditional support and deep trust in me, without whom my project would have been a mere dream rather than a reality.

I would also thank all the academic and administrative staff of School of Mathematics and Computer Applications, Thapar University, Patiala.

Finally, I am also thankful to my friends who also contributed a lot in accomplishing this piece of work.

Isha Rani
(ISHU)

Dated: 15-07-11

ABSTRACT

Network flows are growing interest from the point of view of both application and theory. The topic of network flows has application in such as diverse fields as engineering, management, Science, Computer Science to name but a few. Related to theory, network flow have been proven to be excellent indicator of things to come in mathematical programming.

Maximum flow problem is a classical network flow problem. In general, there are two principal categories for solving the maximum problem, the labelling method and preflow-push method. The labelling method increase the flow along augmentation path from source node to sink node. The idea of preflow-push algorithm is to seek out the shortest path as in the labelling method, but do not send flow along paths from source node to sink node.

Present thesis deals the algorithm for the maximal flow problems. The thesis contains four chapters. Chapter one is introductory in nature in which the terminology needed to discuss the maximal flow problem and brief survey on literature related to the topic have been discussed. In chapter two, “On critical capacity on arcs in a directed network” given by Sonia and Puri (2006) is reviewed. In chapter three, the cut search algorithm with arc capacity and lower bounds given by Phillips and Dessouky (1979) is reviewed. In chapter four, an attempt has been made to apply the algorithm given by Sonia and Puri (2006) to a network in which the network is undirected.

CONTENTS

Chapter 1: INTRODUCTION 1 - 10

Chapter 1

INTRODUCTION

1.1 INTRODUCTION

Network flow problems has caught the attention and stimulated the interest of mathematicians, academicians, operation research analysts, and one of the common name associated with network flow problem is maximal flow problem. Network flow algorithms have been developed to handle a special class of linear programming problems. These problems may each be described by a network whose links carry flow. Network algorithms take advantage of this special structure to produce an optimal solution much more quickly, with less storage required, and with virtually no round off error in comparison with general linear programming codes. In this study we start our discussion by introducing network structured linear programming and some associated concepts relavent to this study followed by the terminology of the networks.

1.2 TERMINOLOGY OF NETWORKS

We encounter many different types of networks in our everyday lives, including electrical, telephone, cable, highway, rail, manufacturing, and many more computer networks. Networks consist of special points called nodes and links connecting pairs of nodes called arcs. Some examples of networks are listed in Figure 1.1. Moreover, because the physical operating characteristics of networks have natural mathematical representations, practitioners and non-specialists can readily understand the mathematical descriptions of network optimization problems and the basic nature of techniques used to solve these problems. This combination of widespread applicability and ease of assimilation has undoubtedly been instrumental in the evolution of network planning models as one of the most widely used modelling techniques in all of operations research and applied mathematics. In all of these networks, some specified commodity which we generically call flow from certain supply points to some demand points, and do so as efficiently as possible, subject to certain constraints. Network flow theory is the study of designing computationally efficient algorithms to solve such problems.

Network	Nodes	Arcs	Flow
Communication	Telephone exchanges,computes , satellites	Cables, fibre optics	Voice, messages, video, data
Hydraulic	Reservoirs, Lakes ,pumping stations	Pipeline	Hydraulic fluid, water, gas, oil
Financial	currencies, stock	Transaction	Money
Transportation	airports, rail yards, intersection	highways, railbeds, airline routes	freight, vehicles, passengers

Figure 1.1

Arc: A line joining points i and j is called an arc if it can only be used in one specified direction, say, from i to j; and it is denoted by the ordered pair (i, j). The arc (i, j) is incident into point j and incident out of point i.

Edge: A line joining two points i and j that can be used either in the direction from i to j, or from j to i, is called an edge and it is denoted by the unordered pair $(i; j)$. The edge $(i; j)$ is said to be incident at points i, j.

A graph is a pair $G = (V, E)$, where $V = V(G)$ is a finite non-empty set and $E = E(G)$ consists of pairs of distinct elements of V. The elements of V are called vertices, the elements of E are called edges. The order of a graph is the number of its vertices, i.e. $|V(G)|$. The size of a graph is the number of its edges, i.e. $|E(G)|$.

Directed network: The network is a called a directed network if all the lines in it are arcs.

Undirected network: The network is a called an undirected network if all the lines in it are edges.

Mixed network: A network that has both arcs and edges is often called a mixed network.

Path: A path (from node i_0 to i_p) is a sequence of arcs $P = \{(i_0, i_1), (i_1, i_2), \ldots, (i_{p-1}, i_p)\}$ in which the initial node of each arc is the same as the terminal node of the preceding arc in the sequence and $i_{0,} \ldots, i_p$ are all distinct nodes. Thus each arc in the path is directed "toward" i_p and "away" from $i_{0.}$

Chain: A chain is a similar structure to a path except that not all arcs are necessarily directed toward node i_p.

Figure 1.2 illustrates a path, and Figure 1.3 presents a chain.

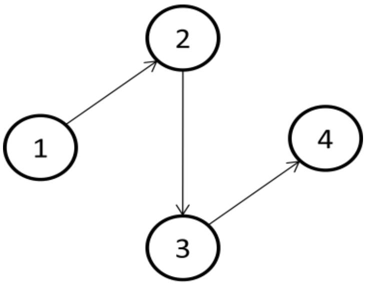


Figure 1.2

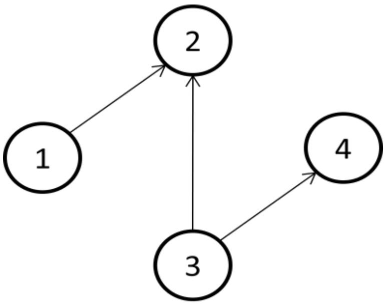


Figure 1.3

Circuit: A circuit is a path from source node i_0 to i_p plus the arc (i_p, i_0).Thus a circuit is a closed path.

Cycle: A cycle is a closed chain.

Every path is a chain but not vice versa. Every circuit is a cycle but not conversely.

Figure 1.4 and 1.5 depict circuits and cycles.

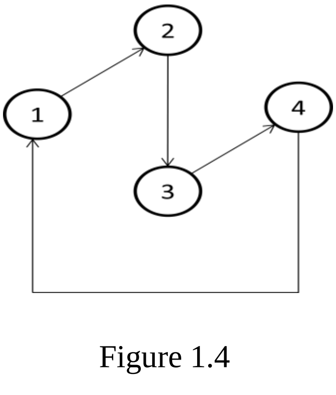


Figure 1.4

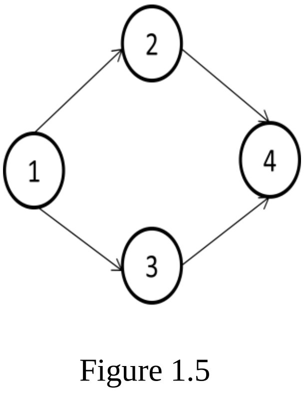


Figure 1.5

Sub-networks and partial networks: A sub-network of G is a network $G' = (N, A')$ with the same set of points, but with $A' \subset A$.

A partial network of G is a network (N', A') in which the set of points is $N' \subset N$, and the set of lines is A', which is the set of all lines in G that have both their incident points in N'.

A partial sub-network of G is a partial network of a sub-network of G.

Degree: The degree of a node is the number of arcs incident at it.

The indegree of a node i is the number of arcs that have i as their to-node, and out-degree of i is the number of arcs that have i as their from node.

Hence (in-degree of i) + (out-degree of i) = (degree of i).

For example, in Figure 1.6 the in-degree of node i is 1 and its out-degree is 2; its degree is 3.

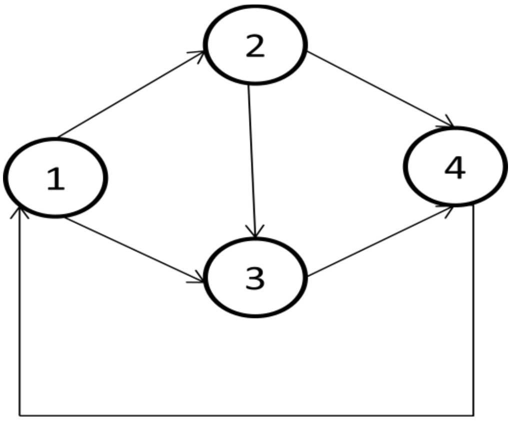


Figure 1.6

Arc flow

Flow is associated with the network, entering and leaving at the nodes and passing through the arcs. The flow in arc k is x_k. Flow is conserved at the nodes, implying that the total flow entering a node must equal the total flow leaving the node.

Upper and lower bounds on flow

Flow is limited in an arc by the lower and upper bounds on flow. Sometimes the term capacity refers to the upper bound on flow. We use l_k and u_k for the lower and upper bounds of arc k. Upper bound represents the maximal flow of some commodity that can pass from node i to node j along arc (i, j) and lower bound is a minimal flow that must be maintained.

Feasible network:

Consider a directed network $G = (N, A)$ comprising of n nodes and m arcs. Arc $(i, j) \in A$ has a lower bound l_{ij} and an upper bound u_{ij} on the arc flow. When $l_{ij} = 0$ for all $(i, j) \in A$, the max flow from one node (say node 1) to the other node (say node n) is always finite. Since the vector zero is itself feasible. In such situations G is said to be a feasible network.

Feasible Flow

A feasible flow is an assignment of flow to the arcs that satisfies conservation of flow for each node and the bounds on flow for each arc.

Definition: Let *N* be a capacitated *s*-*t*-network. A feasible flow *f* in *N* is:

1. (capacity constraints) $f(e) \leq cap(e)$, for every arc *e* in network *N*.
2. (conservation constraints) $\sum_{e \in In(e)} f(e) = \sum_{e \in out(e)} f(e)$.

for every vertex v in network N, other than source s and sink t.

1.3 MAXIMAL FLOW PROBLEM

Consider a network with m nodes and n arcs through which a single commodity flow. We associate with each arc (*i*, *j*) a lower bound on flow of $l_{ij} = 0$ and an upper bound on flow of u_{ij}.we shall assume throughout the development that the u_{ij}'s (arc capacities) are finite integers. There is no cost involved in the maximal flow problem. In such a network, we wish to find the maximum amount of flow from node 1 to node *m*.

Let *f* represent the amount of flow in the network from node 1 to node *m*. Then the maximal flow problem may be stated as follows:

Consider a network with m nodes and n arcs through which a single commodity flow. We associate with each arc (*i*, *j*) a lower bound on flow of $l_{ij} = 0$ and an upper bound on flow of u_{ij}.we shall assume throughout the development that the u_{ij}'s (arc capacities) are finite integers. There is no cost involved in the maximal flow problem. In such a network, we wish to find the maximum amount of flow from node 1 to node *m*.

Let f represent the amount of flow in the network from node 1 to node *m*. Then the maximal flow problem may be stated as follows:

$$\text{Maximize} \qquad f$$

$$\sum_{j=1} x_{ij} - \sum_{k=1} x_{ki} \quad = f \quad \text{if } i = 1 \tag{1.1}$$

$$= -f \quad \text{if } i = \text{m}$$
$$= 0 \quad \text{if } i \neq 1 \text{or } m$$

$$x_{ij} \leq u_{ij}, \; i, j = 1, 2, ..., m$$

$$x_{ij} \geq 0, \; i, j = 1, 2, ..., m$$

Where the sums and inequalities are taken over existing arcs in a network. This is called the node-arc formulation for the maximal flow problem since the constraint matrix noting that *v* is a variable and denoting the node arc incidence matrix by *A*, noting that *f* is a variable and denoting the node-arc incidence matrix by *A*, we can write the maximal flow problem in matrix form as:

$$\text{Maximize} \qquad f$$

$$\begin{aligned} \text{Subject to} \qquad & (e_m - e_1) f - Ax = 0 \\ & x \leq u \\ & x \geq 0 \end{aligned}$$

Since the activity vector for f is $(e_m - e_1)$, the difference of two unit vectors, we may view f as a flow variable on an arc from node *m* to node 1. This provides the direct formulation of the maximal flow problem in (out-of-kilter) circulatory form (with zero right-hand –side values for the flow conservations equations).

Before continuing further it is important to introduce the concept of feasible flow vector and cutsets

1.3.1 FEASIBLE FLOW VECTOR

A flow vector $f = f_{ij}$ specifies the amount of flow on each arc of the network. A feasible flow vector satisfies

$$l_{ij} \le f_{ij} \le k_{ij} \qquad \forall (i,j) \in A$$

Maximize f

$$\sum_{j \in A_i} f_{ij} - \sum_{j \in B_i} f_{ji} = v \qquad \text{if } i \text{ is source} \qquad \cdots(1.1)$$

$=-v$ if i is sink

$= 0$ if i is an intermediate point

Where $A_i = \{j : j \,\text{such that}\, (i,j) \in A\}; B_i = \{j : j \,\text{such that}\, (j,i) \in A\}$ and v, the net amount of material leaving the source (and arriving at the sink) is known as the value of the flow vector *f*. The constraints (1.1) are known as flow conservation equations. They require that the total amount of material reaching should be equal to the total amount of material leaving, at every intermediate point, e.g. for an oil pipeline, the total flow of oil going into any juncture (vertex) in the pipeline must equal the total flow leaving that juncture.

Where the sums and inequalities are taken over existing arcs in a network. This is called the node-arc formulation for the maximal flow problem since the constraint matrix noting that *v* is a variable and denoting the node arc incidence matrix by *A*, noting that *f* is a variable and denoting the node-arc incidence matrix by *A*, we can write the maximal flow problem in matrix form as:

Maximize v

Subject to
$$(e_m - e_1)v - Af = 0$$
$$f \le u$$
$$f \ge 0$$

Since the activity vector for v is $(e_m - e_1)$, the difference of two unit vectors, we may view v as a flow variable on an arc from node m to node 1. This provides the direct formulation of the maximal flow problem in (out-of-kilter) circulatory form (with zero right-hand-side values for the flow conservations equations).

Before continuing with the development of an algorithm to solve this maximal flow problem, we introduce the useful and important concept of cuts.

1.3.2 FLOW AUGMENTING PATHS

Let *G* be a directed network with capacity vector *u* and lower bound *l*. Let f be the feasible flow vector on *G*. A path *P*, from the source to the sink is said to be a flow augmenting path (FAP) with respect to *f* if it satisfies

$$f_{ij} < u_{ij} \qquad \text{whenever } (i, j) \text{ is a forward arc on the path } P$$

$$f_{ij} > l_{ij} \qquad \text{whenever } (i, j) \text{ is a reverse arc on the path } P$$

The reason for this name can be easily explained. Let *v* be the value of the flow vector *f*. Let

$$\varepsilon_1 = \text{minimum}\{(u_{ij} - f_{ij}) : (i, j) \text{ a forward arc on } P\}$$

$$= +\infty, \text{if all the arcs on } P \text{ are reverse arcs}$$

$$\varepsilon_2 = \text{minimum}\{(f_{ij} - l_{ij}) : (i, j) \text{ a reverse arc on } P\}$$

$$= +\infty, \text{if all the arcs on } P \text{ are forward arcs}$$

$$\varepsilon = \text{minimum}\{\varepsilon_1, \varepsilon_2\}$$

ε is obviously positive. Define a new flow vector $\hat{f}$ by

$$\hat{f}_{ij} = f_{ij} + \varepsilon \qquad \text{if } (i, j) \text{ is not on the path } P$$

$$= f_{ij} + \varepsilon \qquad \text{if } (i, j) \text{ is a forward arc on } P$$

$$= f_{ij} - \varepsilon \qquad \text{if } (i, j) \text{ is } a \text{ reverse arc on } P$$

Then $\hat{f}$ is a feasible flow vector whose value is $\hat{v} = v + \varepsilon$ .Hence, given a feasible flow vector and an FAP with respect to it, the FAP can be used to construct another feasible flow vector with a higher value.

1.3.3 CUT-SET (SEPARATING NODE m FROM NODE 1)

Let X be any set of nodes in the network such that X contains node 1 but node m. Let $\overline{X} = (N - X)$. Then $(X, \overline{X}) = \{(i, j) : i \in X, j \in \overline{X}\}$ is called a cut-set separating node m from node 1.

1.3.4 CAPACITY OF A CUT-SET

Let $(X, \overline{X})$ be any cut-set in a network G. Then $u(X, \overline{X}) = \sum_{(i,j)\in(X,\overline{X})} u_{ij}$ is called the capacity of the cut-set. In Figure 1.7 there are several cut-sets separating node 4 from node 1 in G. They are

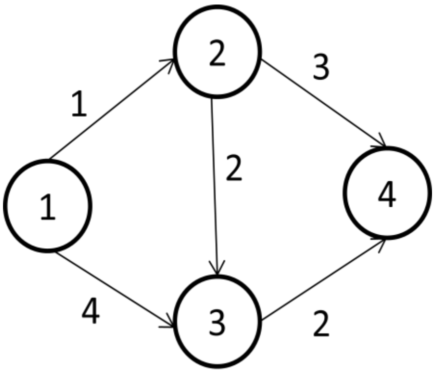


Figure 1.7

$X = \{1\}$ $\quad \overline{X} = \{2,3,4\}$ $\quad (X, \overline{X}) = \{(1,2),(1,3)\}$ $\quad u(X, \overline{X}) = 5$

$X = \{1,2\}$ $\quad \overline{X} = \{3,4\}$ $\quad (X, \overline{X}) = \{(1,3),(2,3),(2,4)\}$ $\quad u(X, \overline{X}) = 9$

$X = \{1,3\}$ $\quad \overline{X} = \{2,4\}$ $\quad (X, \overline{X}) = \{(1,2),(3,4)\}$ $\quad u(X, \overline{X}) = 3$

$X = \{1,2,3\}$ $\quad \overline{X} = \{4\}$ $\quad (X, \overline{X}) = \{(2,4),(3,4)\}$ $\quad u(X, \overline{X}) = 5$

1.3.5 CUTSET SEPARATING THE SOURCE AND THE SINK

Let *G* be a directed network. Let *X* be a subset of points of *G* containing the source and not containing the sink. Let $\overline{X}$ be the set of all the points of G that are not in *X*. The partition *X*, $\overline{X}$ generates a cutest separating the source and the sink. The set of forward arcs of this cutest is $\{(i,j):(i,j)\in A, i\in X, j\in \overline{X}\}$. The set of reverse arcs of this cutest is $\{(i,j):(i,j)\in A, i\in \overline{X}, j\in X\}$. The cutest itself is denoted by $(X, \overline{X})$.

The capacity of the cutest $(X, \overline{X})$ separating the source and the sink is defined as:

$$\sum_{\substack{(i,j) \\ \textit{forward arc}}} k_{ij} - \sum_{\substack{(i,j) \\ \textit{reverse arc}}} l_{ij}$$

1.4 GENERALIZED MAXIMUM FLOW PROBLEM

In this dissertation, we consider a network flow problem called the generalized maximum flow problem. First, we describe the traditional maximum flow problem. This problem was first studied by Dantzig and Ford and Fulkerson in the 1950's. The problem is simple to state and is defined formally in given capacity limits on the arcs, the goal is to send as much flow as possible from one distinguished node called the source to another called the sink. For example, a power company may wish to maximize the amount of natural gas sent between a pair of cities through its network of pipelines. Each pipeline in the network has a limited capacity.

The generalized maximum flow problem is a natural generalization of the traditional maximum flow problem. It was first investigated by Dantzig and Jewell in the 1960's. In traditional networks, there is an implicit assumption that flow is conserved on every arc. This assumption may be violated if natural gas leaks as it is pumped through a pipeline. The generalized maximum flow problem generalizes the traditional maximum flow problem by allowing flow to “leak" as it is sent through the network. As before, each arc (v, w) has a capacity $u(v;\mathrm{i})$ that limits the amount of flow sent into that arc.

LITERATURE SURVEY

Network optimization experienced a fast development, during the last few decades. Spurred by Ford-Fulkerson's seminal work in the 60s, studied in this area have led to continued improvements in efficient network flows algorithms. This is especially in respect to polynomial time algorithm for shortest path, maximal flows and minimum cost flows that underwent intensive development in 80s to 90s. There have been a number of survey papers and books that have been reported on network flows. For example Ahuja (1992), Asano and Asano (1997), Bertsekas (1998), Cook and Canninghan (1998), Kapoor and Vaidya (1996), Murray (1993), Schrijver (2003) and Shigeno (2003) published the work on network flows.

Ford and Fulkerson (1956) and dantzig and Fulkerson (1956) have been given the proof of the theorem that the maximum possible flow from left to right through a network is equal to the minimum value among a simple cut-sets. This theorem may appear almost obvious on physical grounds and appears to have been accepted without proof for some time by workers in communication theory. However, while the fact that this flow can not be exceeded is indeed almost trivial, The fact that it can actually be achieved is by no means obvious.

Dantzig, Ford and Fulkerson (1950) proposed the maximum flow problem. This problem is simple to state given capacity limits on the arcs, the goal is to send as much flow as possible from one distinguished node called the source to another called the sink. For example, a power company may wish to maximize the amount of natural gas sent between a pair of cities through its network of pipelines. Each pipeline in the network has a limited capacity.

Swanson and Woolsey (1973) proposed a network flow algorithms to handle a special class of linear programming problems. These problems may each be described by a network whose links carry flow. Network algorithms take advantage of this special structure to produce an optimal solution much more quickly, with less storage required, and with virtually no round off error in comparison with general linear programming codes. The objective of this paper is to introduce the reader to the basic ideas involved in formulating a network problem, and to explain in a non-technical manner what the "out-of-kilter" network algorithm does. This algorithm is very general and extremely efficient in solving network flow cost problems (as opposed to CPM, PERT, and GERT type networks).

Dantzig and Jewell (1960) proposed the generalized maximum flow problem is a natural generalization of the traditional maximum flow problem.

Wayne (1999) proposed a generalized maximum flow algorithm which present several new efficient algorithms for the generalized maximum flow problem. In the traditional maximum flow problem, there is a capacitated network and the goal is to send as much of a single commodity as possible between two distinguished nodes, without exceeding the arc capacity limits. The problem has hundreds of applications including: shipping freight in a transportation network and pumping fluid through a hydraulic network.

Ghiyasvand (2006) proposed a new approach for computing a most positive cut using the minimum flow algorithms which describes the correspondence between the minimum flow problem and the most positive cut problem. A new algorithm is presented to solve the most positive cut problem using the minimum flow algorithms and it has been proved that the infeasibility of a network can be distinguished by solving a minimum flow problem.

In a generalized network model each arc (i, j) of the network has a positive multiplier $\gamma(i, j)$, which implies that if we send one unit from the node i to the node j along the arc (i, j), then $\gamma(i, j)$ units arrived at the node j. The multiplier is called a gain. This network model arises in manufacturing, transportation, communication and financial analysis. A problem finding a maximum flow in the generalised network is called a generalized maximum flow problem.

Since Dantzig (1962) and Jewell (1962) did their first studies in the 1960s, a number generalized maximum flow algorithm has been developed. The problem itself is a speial case of the linear programming problem. Consequently, Dantzig (1962) proposed a generalized network simplex algorithm, which is specialization of the simplex method. Murray (1993) also design an interior point algorithm for generalized flow. Goldfarb-Lin (2002) presented an interior point algorithm with a combinational method, i.e., flow augmentations. This interior-point algorithm can be applied to generalize minimum cost flows.

Present work:

Present thesis deals the algorithm for the maximal flow problems. The thesis contains four chapters. Chapter one is introductory in nature in which the terminology needed to discuss the maximal flow problem and brief survey on literature related to the topic have been discussed. In chapter two, “On critical capacity on arcs in a directed network” given by Sonia and Puri (2006) is reviewed. In chapter three, the cut search algorithm with arc capacity and lower bounds given by Phillips and Dessouky (1979) is reviewed. In chapter four, an attempt has been made to apply the algorithm given by Sonia and Puri (2006) to a network in which the network is undirected.

Chapter 2

ON CRITICAL CAPACITY OF ARCS IN A DIRECTED NETWORK

2.1. INTRODUCTION

A methodology proposed by Sonia and Puri (2006) to find the critical capacity of an arc of a feasible network wherein lower bounds on flows on the various arcs are non-negative integers is reviewed in this chapter. The procedure given by Sonia and Puri (2006) is presented in algorithm form.

A directed network $G=(N,A)$ having $n=(|N|)$ nodes and $m=(|A|)$ arcs. Each arc $(i,j)\in A$ has a lower and upper bound which are denoted by l_{ij} and u_{ij}. When $l_{ij}=0,$ there is always a feasible flow vector. Since the vector zero is itself feasible, the max flow from one node to other node is always finite. Then the network G is called feasible network.

2.2. CRITICAL CAPACITY

The critical capacity is defined as the extent up to which flow can be increased on the arc then the value of max flow increases and denoted by k_{ij}^{*} of an arc $(i=i_1,i_2,i_3,\dots;\ j=j_1,j_2,j_3,\dots)$ and it is formulated by the formula $k_{ij}^{*}=v(\infty)-v(0)$. If $v(0)$ is the max flow value from the node 1 (source) to the node n (sink) when the flow on the arc (i_1,j_1) is set at 0 and $v(\infty)$ is the value of the max flow when the flow on the arc (i_1,j_1) is set at ∞ and the capacity of arcs except (i_1,j_1) set equals to upper bound. That is, as the capacity of the arc (i_1,j_1) is raised from 0 to k_{i_1,j_1}^{*} the value of the max flow from the source to the sink rises from $v(0)$ to $v(\infty)=v(0+k_{i_1,j_1}^{*})$. Further, raising the flow on (i_1,j_1) beyond k_{i_1,j_1}^{*} will not raise the value of max flow. Similarly, we find the critical capacities of all arcs. Thus, the max flow from the source to the sink.

$$=v(0+\lambda);\qquad 0\le\lambda\le k_{ij}^{*}$$

$$=v(0+k_{ij}^{*})=v(\infty);\qquad \lambda>k_{ij}^{*}$$

Further, whenever the capacity of the arc (i,j) is :

(a) $k_{ij} < k^{*}{}_{ij}$, (i, j) is a forward arc in every minimum capacity cutest separating the source and the sink.

(b) $k_{ij} > k^{*}{}_{ij}$, (i, j) is not a forward arc in any minimum capacity cutest separating the source and the sink.

(c) $k_{ij} = k^{*}{}_{ij}$, (i, j) there exists a minimum capacity cutest separating the source and the sink that contains (i, j) as a forward arc, and another that does not contain (i, j) as a forward arc .It is also known that the destruction of an arc (i, j) reduces the max flow value from the source to the sink by $\min(k_{ij}, k^{*}{}_{ij})$,where destroying an arc (i, j) means setting $u_{ij} = l_{ij} = 0$. When $l_{ij} \geq 0, \forall (i, j) \in A$ is examined by solving a circulation problem created by linking the sink node to the source node by the arc (n, 1) with lower bound $l_{n1} = 0$ and upper bound $u_{n1} = \infty$ on the arc flow. Existence of a feasible circulation is examined by finding the maximum flow from the new source s^{*} to the new sink t^{*} in the enlarged network $G^{*} = (N^{*}, A^{*})$. where $N^{*} = N \cup \{s^{*}, t^{*}\}$ and $A^{*} = A \cup \{(n,1)\} \cup \{(s^{*}, i), (i, t^{*}), i \in N\}$.

The flow on the arc (i, j) in A^{*} has lower bound l^{*}_{ij} and upper bound u^{*}_{ij} defined as:

(a) $l^{*}_{ij} = 0$ for all $(i, j) \in A^{*}$

(b) $u^{*}_{ij} = u_{ij} - l_{ij}$ for all $(i, j) \in A$

(c) $u^{*}_{s^{*}i} = \sum_{\substack{r \in N: \\ (r,i) \in A}} l_{ri},\ i \in N$

(d) $u^{*}_{it^{*}} = \sum_{\substack{r \in N: \\ (i,r) \in A}} l_{ir},\ i \in N$

(e) $u^{*}_{n1} = \infty$

Let the value of the max flow is V. If $V < \sum_{(i,j) \in A} l_{ij}$, then there does not exist a feasible circulation and G is termed as infeasible.

If $V = \sum_{(i,j) \in A} l_{ij}$, then there exists a feasible circulation and G is said to be a feasible network in the sense it will have a finite max flow from the source to the sink. In this case, arcs (s^{*}, i) and $(i, t^{*}), i \in N$ are saturated and there are only two minimum capacity cutsets separating the source and the sink. Therefore, arcs of G will not be forward arcs of the minimum capacity cutsets separating the source and sink.

2.3 THEORETICAL DEVELOPMENT

Let $G=(N,A)$ be a directed network and $l_{ij}\geq 0$, u_{ij}, respectively be the lower bound and the upper bound on the flow on an arc $(i,j)\in A$

Let $v(l_{ij}+\lambda)$ be the max flow in G from the source to the sink when the upper bound on the flow on the arc (i,j) is set equals to $(l_{ij}+\lambda)$

Definition 2.3.1 (Modified lower bound).Suppose $v(l_{ij}+\lambda)$ does not exist for $0\leq\lambda<L_{ij}$.If $v(l_{ij}+L_{ij})$ exists, then $\hat{l}_{ij}=l_{ij}+L_{ij}$ is called the modified lower bound on the flow on the arc (i,j).

Lemma: Whenever the max flow in G^* is $\sum_{(i,j)\in A} l_{ij}$, there exist ONLY two minimum capacity cutsets separating the source and the sink namely:

(i) $(X,\overline{X})$, where $=\{s^*\}, \bar{X}=N^*\setminus\{s^*\}$ and

(ii) $(X,\overline{X})$, where $=N^*\setminus\{t^*\}$, $\overline{X}=\{t^*\}$.

Proof: It is given that the max flow value from the source s^* to the sink t^* in G^* is $\sum_{(i,j)\in A} l_{ij}$. This means that the net flow out of the source s^* and the net flow into the sink t^* is $\sum_{(i,j)\in A} l_{ij}$.

As $\sum_{i\in N} u^*_{s^*i}=\sum_{i\in N}(\sum_{\substack{r\in N:\\(r,i)\in A}} l_{ri})=\sum_{(i,j)\in A} l_{ij}$, the net flow out of the source s^* would be $\sum_{(i,j)\in A} l_{ij}$ if and only if $f^*_{s^*i}=u^*_{s*i}$, $\forall i\in N$ where $f^*_{s^*i}$ is the flow on arc (s^*,i) in the max flow vector. This implies that all the arcs (s^*,i), $i\in N$ going out of the source s^* are saturated.

Similarly, as $\sum_{i\in N} u^*_{it^*}=\sum_{i\in N}(\sum_{\substack{r\in N\\(i,r)\in A}} l_{ir})=\sum_{(i,j)\in A} l_{ij}$, all the arcs (i,t^*) would also be saturated.

This implies that the two cut-sets: $(i)\ (X,\bar{X}), X=\{s^*\}, \overline{X}=N^*\setminus\{s^*\}$ and $(ii)\,(X,\bar{X})$, $X=N^*\setminus\{t^*\}, \overline{X}=\{t^*\}$ have capacity $\sum_{(i,j)\in A} l_{ij}$ and therefore, these are the minimum capacity cut-sets.

Consider any other cutsets $(i)\ (X,\bar{X}), X=\{s^*,i\}, \overline{X}=N^*\setminus X$. This implies that according to the labelling technique if the label on the source node s^* is $(s,+)$, then the label on the node i would be $(s^*,+)$. This means that the flow, say $f^*_{s^*i}$ on (s^*,i) is strictly less than $u^*_{s^*i}$. In this case the net flow out of s^* would be $\sum_{\substack{j\in N\\ j\neq i}} u^*_{s^*j} + f^*_{s^*i}$ which is strictly less than $\sum_{(i,j)\in A} l_{ij} (=\sum_{j\in N} u^*_{s^*j})$. But this is not true. Same will be the outcome when the set X contains some (but not all) other nodes belonging to N in addition to the source s^*.

Hence, when the max flow value in G^* from the source s^* to the sink t^* is $\sum_{(i,j)\in A} l_{ij}$ then there exist exactly two minimum capacity cutsets:

(i) $(X,\overline{X}), X=\{s^*\}, \overline{X}=N^*\setminus\{s^*\}$

(ii) $(X,\overline{X}), X=N^*\setminus\{t^*\}, \overline{X}=\{t^*\}$

Theorem 2.1. Let $G=(N,A)$ be a feasible directed network with bounds on the flow on the arc $(i,j)\in A$ as l_{ij}, u_{ij}. Let k^*_{ij} be the critical capacity of the arc (i,j) in the network G^*. Then modified lower bound on the arc flow on (i,j) is $\hat{l}_{ij}=l_{ij}+k^*_{ij}$.

Proof. As $l^*_{ij}=0\ \forall (i,j)\in A^*, G^*$ is a feasible network and hence max flow vector in G^* will exist. Let max flow from s^* to t^* be V. As G is given to be a feasible network, $V=\sum_{(i,j)\in A} l_{ij}$ and a minimum capacity cutset separating the source and the sink with respect to the max flow vector in G^* would be $(X,\bar{X})$, where $X=\{s^*\}, \overline{X}=N^*\setminus\{s^*\}$ or $X=N^*\setminus\{t^*\}, \overline{X}=\{t^*\}$. Therefore, arcs of G will not be forward arcs of the minimum capacity cutsets separating the source and the sink [Lemma]. Therefore, critical capacity of an arc $(i,j)\in A$ will not be more than its current capacity $k_{ij}=u^*_{ij}-0=u_{ij}-l_{ij}$. Destruction of an arc $(i,j)\in A$ will reduce the max flow from s^* to t^* in G^* by $\min(k_{ij}, k^*_{ij}) = k^*_{ij}$. This implies that if $k^*_{ij} > 0, (i, j) \in A$ then its destruction will reduce the max flow from s^* to t^* in G^* *to* ($\sum_{(i,j)\in A} l_{ij} - k^*_{ij}) < \sum_{(i,j)\in A} l_{ij}$, This means that G becomes infeasible on destruction of $(i, j) \in A$ for which $k^*_{ij} > 0$. If $k^*_{ij} = 0, (i, j) \in A$, then destruction of (i, j) will not reduce the max flow in G^* from s^* to t^* below $\sum_{(i,j)\in A} l_{ij}$ and hence G will remain feasible.

Thus, it follows that if the critical capacity of an arc of A is positive with respect to G^*, then its lower bound should be modified. That is, if $k^*_{ij} > 0$ for an arc $(i, j) \in A$, then its modified lower bound $\hat{l}_{ij}$ is given by $\hat{l}_{ij} = l_{ij} + k^*_{ij} (\le u_{ij})$.

Hence, $\hat{l}_{ij} = l_{ij} + k^*_{ij} \ \forall (i, j) \in A$, where k^*_{ij} is the critical capacity of (i, j) with respect to G^*.

Definition 2.3.2 : (Critical capacity of an arc $(i, j) \in A$, when $l_{ij} \ge 0$). If the modified bounds on an arc $(i, j) \in A$ $\hat{l}_{ij}$ and u_{ij} ,its critical capacity is defined as the extent up to which the flow on the arc can be increased beyond $\hat{l}_{ij}$ inducing the raise in the max flow from the source node to the sink node.Thus, if $\hat{k}_{ij}$ is the critical capacity of arc (i, j) ,then raising the flow on (i, j) beyond $\hat{l}_{ij}$ will raise the max flow from the source to the sink in G and it will rise up to $v(\hat{l}_{ij} + \hat{k}_{ij})$ .Thereafter, max flow will not be affected.

Thus Max flow in

$$G = v(\hat{l}_{ij} + \lambda) \forall \lambda : \ 0 \le \lambda \le \hat{k}_{ij}$$

$$= v(\hat{l}_{ij} + \hat{k}_{ij}) \forall \lambda : \ \lambda > \hat{k}_{ij}$$

Notice that the current capacity of $(i, j) \in A$ is $k_{ij} = u_{ij} - \hat{l}_{ij}$

Remark: If the $k_{ij} < \hat{k}_{ij}$, then raising the upper bound on flow on arc (i, j) from $\hat{l}_{ij}$ to $(\hat{l}_{ij} + \hat{k}_{ij})$ will raise the max flow and therefore, this arc must be a forward arc in every minimum capacity cutest.

Remark : If there is a forward arc in every minimum capacity cutest (separating the source and the sink) joining source to sink directly,then critical capacity of such an arc is infinitely large.

2.4 ALGORITHM:

Step 1: Consider a directed network $G = (N, A)$ having $n = (|N|)$ nodes and $m = (|A|)$ arcs in which each arc has lower and upper bound which are denoted by l_{ij} and u_{ij}. $l_{ij} \le u_{ij}$ always.

Step 2: Check the condition for the feasibility of the network, *i.e.* the flow amount from source to sink is to be finite or not with the following procedure:

(i) If $l_{ij} = 0,$ there is always a feasible flow vector. Since the vector zero is itself feasible. The network is a feasible network.

(ii) If $l_{ij} \ge 0,$ and integer for all $(i, j) \in A,$ then the maximum flow in network from source to sink may not exist. So network may not be a feasible network. In this case the feasibility of` network *G* when $l_{ij} \ge 0, \forall (i, j) \in A$ is examined by solving a circulation problem created by linking the sink node to the source node by the arc (*n*, 1) with lower bound $l_{n1} = 0$ and upper bound $u_{n1} = \infty$ on the arc flow.

Step 3: Existence of a feasible circulation is examined by finding the maximum flow from the new source s^* to the new sink t^* in the enlarged network $G^* = (N^*, A^*)$.

Where $N^* = N \cup \{s^*, t^*\}$ and $A^* = A \cup \{(n,1)\} \cup \{(s^*, i), (i, t^*), i \in N\}$

Step 4: The flow on the arc (*i*, *j*) in A^* has lower bound l_{ij}^* and upper bound u_{ij}^* defined as:

(a) $l_{ij}^* = 0$ for all $(i, j) \in A^*$

(b) $u_{ij}^* = u_{ij} - l_{ij}$ for all $(i, j) \in A$

(c) $u_{s^*i}^* = \sum_{\substack{r \in N: \\ (r,i) \in A}} l_{ri}, \ i \in N$

(d) $u_{it^*}^* = \sum_{\substack{r \in N: \\ (i,r) \in A}} l_{ir}, \ i \in N$

(e) $u_{n1}^* = \infty$

Step 5: Determine the max flow from s^* to t^* in G^* . Let the value of the max flow from new source to new sink is *V*. Then it follows the conditions:

(a) If $V < \sum_{(i,j)\in A} l_{ij}$, then there does not exist a feasible circulation and *G* is termed as infeasible.

(b) If $V = \sum_{(i,j)\in A} l_{ij}$, then there exists a feasible circulation and *G* is said to be a feasible network in the sense it will have a finite max flow from the source to the sink.

In this case, arcs (s^*, i) and $(i, t^*), \forall i \in N$ are saturated and there are only two minimum capacity cutsets separating the source and the sink. Therefore, arcs of G will not be forward arcs of the minimum capacity cutsets separating the source and sink.

Step 6: To determine the critical capacity k^*_{ij} of each arc $(i,\ j) \in A$ in the network G^* by using $k^*_{ij} = v(\infty) - v(0)$, where

$v(0)$ is the max flow value from node 1 (source) to the node n (sink) when the flow on the arc (i, j) is set at $l_{ij} = 0$ and the flow on other arcs except (i, j) set equals to upper bound.

$v(\infty)$ is the value of the max flow when the flow on the arc (i, j) is allowed up to any large extent and the flow on other arcs except (i, j) set equals to upper bound.

Step 7: To find the modified lower bound on the flow on each arc (i, j) in the original network G using critical capacities in the following way:

(i)If $k^*_{ij} > 0, (i, j) \in A$ then its destruction will reduce the max flow from s^* to t^* in G^* *to* $(\sum_{(i,j)\in A} l_{ij} - k^*_{ij}) < \sum_{(i,j)\in A} l_{ij}$ (This means that *G* becomes infeasible on destruction of $(i, j) \in A$ for which $k^*_{ij} > 0$. Thus, it follows that if the critical capacity of an arc of *A* is positive with respect to G^*, then its lower bound should be modified) and the modified lower bound $\hat{l}_{ij}$ is given by $\hat{l}_{ij} = l_{ij} + k^*_{ij} (\le u_{ij})$.

(ii) If $k_{ij}^{*}=0, (i, j) \in A,$ (then destruction of (i, j) will not reduce the max flow in G^{*} from s^{*} to t^{*} below $\sum_{(i,j) \in A} l_{ij}$ and hence G will remain feasible).The modified lower bound $\hat{l}_{ij}$ is given by $\hat{l}_{ij} = l_{ij}, (i, j) \in A$

Therefore, critical capacity of an arc $(i, j) \in A$ will not be more than its current capacity $k_{ij} = u_{ij} - 0 = u_{ij} - l_{ij}$. Destruction of an arc $(i, j) \in A$ will reduce the max flow from source to sink by $\min(k_{ij}, k_{ij}^{*}) = k_{ij}^{*}$.

Step 8: Find the flow on augmented network in such a way that flow on arc which are incident out of s^{*} and incident in t^{*} are saturated and flow on $(i, j) \in A$ hold conservation at every node of G .

Step 9: Construct a modified network $G = (N, A)$ having $n = |N|$ nodes and $m = |A|$ arcs and has an each arc has modified lower bound and upper bound which are denoted by $\hat{l}_{ij}$ and u_{ij} .

Step 10: To determine the critical capacity $\hat{k}_{ij}$ of an arc $(i, j) \in A$ of modified network G by using $\hat{k}_{ij} = v(\infty) - v(\hat{l}_{ij})$.(Then raising the flow on (i, j) beyond $\hat{l}_{ij}$ will raise the max flow from the source to the sink in G and it will rise up to $v(\hat{l}_{ij} + \hat{k}_{ij})$ .Thereafter, max flow will not be affected), where $v(\hat{l}_{ij})$ is the max flow value from node 1 (source) to the node n (sink) when the flow on the arc (i, j) is set at $\hat{l}_{ij}$ and $v(\infty)$ is the value of the max flow when the flow on the arc (i, j) is allowed up to any large extent.

Step 11: Find the current capacity k_{ij} of modified network G that is $k_{ij} = u_{ij} - \hat{l}_{ij}$

Step 12: Using the procedure given below determine which arc is a forward arc or not

(a) If $k_{ij} < \hat{k}_{ij}$, then raising the upper bound on flow on arc (i, j) from $\hat{l}_{ij}$ to $(\hat{l}_{ij} + \hat{k}_{ij})$ will raise the max flow and therefore, this arc is a forward arc in every minimum capacity cutest separating the source and the sink.

(b) If $k_{ij} > \hat{k}_{ij}, (i, j)$ is not a forward arc in any minimum capacity cutest separating the source and the sink.

(c) If $k_{ij} = \hat{k}_{ij}, (i, j)$ there exists a minimum capacity cutest separating the source and the sink that contains (i, j) as a forward arc, and another that does not contain (i, j) as a forward arc .

Step 13: Find the maximum flow in modified network G by

$$v(\hat{l}_{ij} + \lambda)\forall\lambda : \; 0 \le \lambda \le \hat{k}_{ij}$$

$$v(\hat{l}_{ij} + \hat{k}_{ij})\forall\lambda : \; \lambda > \hat{k}_{ij}$$

Numerical Examples

1. Consider the directed network G in Figure 2.1 with the lower and upper bound on the flow on the arc (i, j) given as (l_{ij}, u_{ij}) .

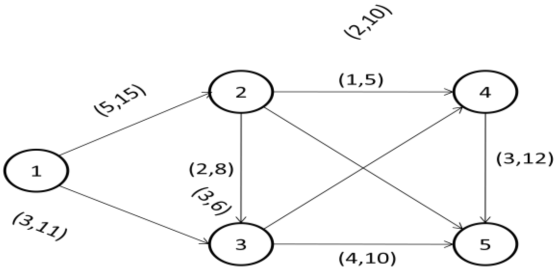


Figure 2.1

Introduce the arc (5,1) with the lower bound on the flow as zero and upper bound as infinity. Using step 2 to step 5, feasibility conditions is checked and the procedure is given from Figure 2.2 to Figure 2.10

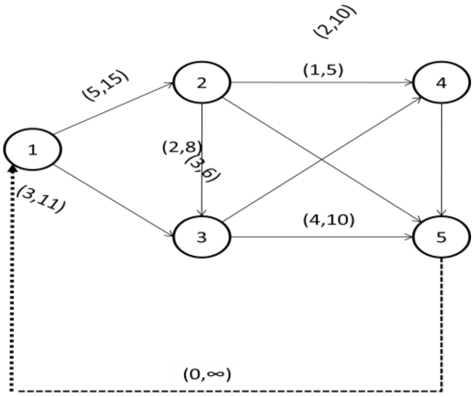


Figure 2.2

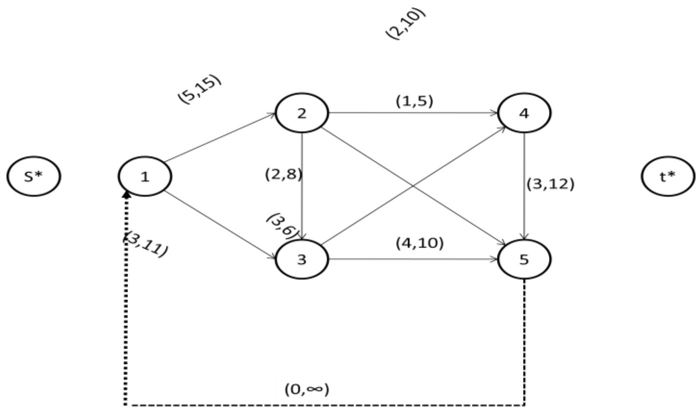


Figure 2.3

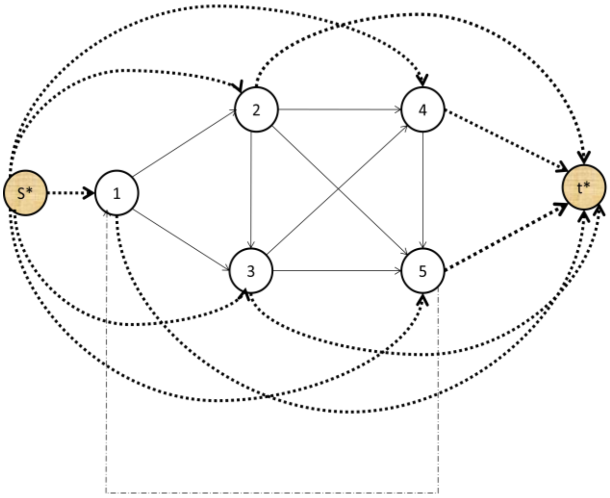


Figure 2.4

Using step 4(a), $l^*_{ij}, \forall(i,j) \in A^*$ shown in Figure 2.5

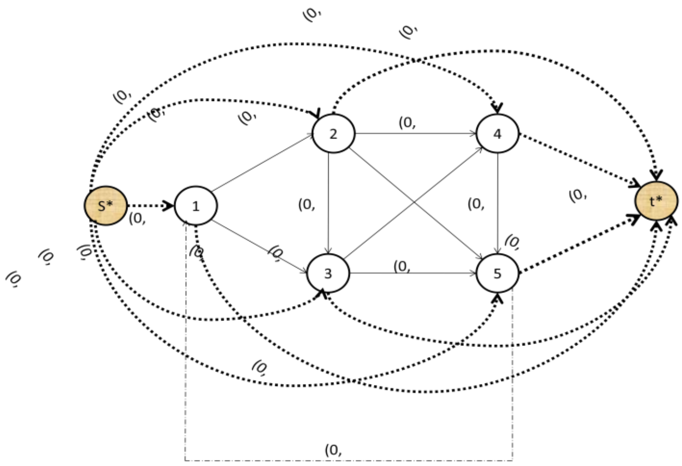


Figure 2.5

Using step 4(b), find the value of $u^*_{ij}, \forall (i, j) \in A$ are given in Table 2.1 and shown in Figure 2.6

Table 2.1

Take $(1,2)$ arc	Take $(1,3)$ arc	Take $(2,3)$ arc	Take $(2,4)$ arc	Take $(2,5)$ arc	Take $(3,4)$ arc	Take $(3,5)$ arc	Take $(4,5)$ arc
$u^*_{12} = u_{12} - l_{12}$	$u^*_{13} = u_{13} - l_{13}$	$u^*_{23} = u_{23} - l_{23}$	$u^*_{24} = u_{24} - l_{24}$	$u^*_{25} = u_{25} - l_{25}$	$u^*_{34} = u_{34} - l_{34}$	$u^*_{35} = u_{35} - l_{35}$	$u^*_{45} = u_{45} - l_{45}$
$u^*_{12} = 15 - 5$	$u^*_{13} = 11 - 3$	$u^*_{23} = 8 - 2$	$u^*_{24} = 5 - 1$	$u^*_{25} = 6 - 3$	$u^*_{34} = 10 - 2$	$u^*_{35} = 10 - 4$	$u^*_{45} = 12 - 3$
$u^*_{12} = 10$	$u^*_{13} = 8$	$u^*_{23} = 6$	$u^*_{24} = 4$	$u^*_{25} = 3$	$u^*_{34} = 8$	$u^*_{35} = 6$	$u^*_{45} = 9$

Table 2.2

(l^*_{12}, u^*_{12})	(l^*_{13}, u^*_{13})	(l^*_{23}, u^*_{23})	(l^*_{24}, u^*_{24})	(l^*_{25}, u^*_{25})	(l^*_{34}, u^*_{34})	(l^*_{35}, u^*_{35})	(l^*_{45}, u^*_{45})
$(0,10)$	$(0,8)$	$(0,6)$	$(0,4)$	$(0,3)$	$(0,8)$	$(0,6)$	$(0,9)$

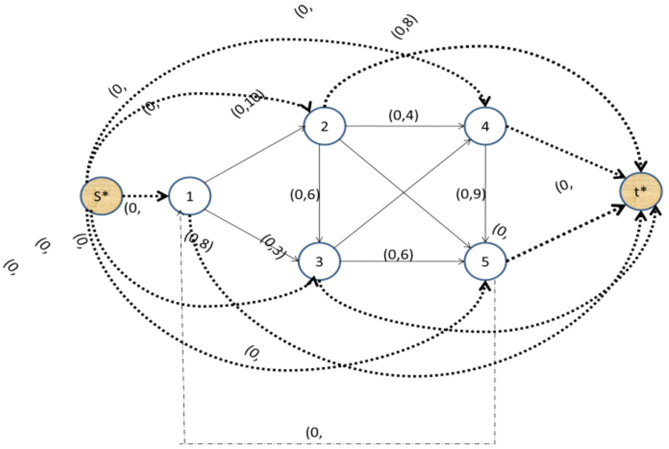


Figure 2.6

Using step 4(c), find the value of $u^*_{s^*i}, i \in N$ are given in Table 2.3 and shown in Figure 2.7

Table 2.3

$i=1$	$i=2$	$i=3$	$i=4$	$i=5$
$u^*_{s^*1} = \sum_{\substack{r\in(1,2,3,4,5)\\(r,1)\in A}} l_{r1}$	$u^*_{s^*2} = \sum_{\substack{r\in(1,2,3,4,5)\\(r,2)\in A}} l_{r2}$	$u^*_{s^*3} = \sum_{\substack{r\in(1,2,3,4,5)\\(r,3)\in A}} l_{r3}$	$u^*_{s^*4} = \sum_{\substack{r\in(1,2,3,4,5)\\(r,4)\in A}} l_{r4}$	$u^*_{s^*5} = \sum_{\substack{r\in(1,2,3,4,5)\\(r,5)\in A}} l_{r5}$
$u^*_{s^*1} = l_{11} + l_{21}$ $l_{31} + l_{41} + l_{51}$	$u^*_{s^*2} = l_{12} + l_{22} +$ $l_{32} + l_{42} + l_{52}$	$u^*_{s^*3} = l_{13} + l_{23} +$ $l_{33} + l_{43} + l_{53}$	$u^*_{s^*4} = l_{14} + l_{24} +$ $l_{34} + l_{44} + l_{54}$	$u^*_{s^*5} = l_{15} + l_{25} +$ $l_{35} + l_{45} + l_{55}$
Only $(5,1) \in A$	Only $(1,2) \in A$	$(1,3),(2,3) \in A$	$(2,4),(3,4) \in A$	$(2,5),(3,5),(4,5) \in A$
$u^*_{s^*1} = l_{51}$	$u^*_{s^*2} = l_{12}$	$u^*_{s^*3} = l_{13} + l_{23}$	$u^*_{s^*4} = l_{24} + l_{34}$	$u^*_{s^*5} = l_{25} + l_{35} + l_{45}$
		$u^*_{s^*3} = 3+2$	$u^*_{s^*4} = 1+2$	$u^*_{s^*5} = 3+4+3$
$u^*_{s^*1} = 0$	$u^*_{s^*2} = 5$	$u^*_{s^*3} = 5$	$u^*_{s^*4} = 3$	$u^*_{s^*5} = 10$

Table 2.4

$(l^*_{s^*1}, u^*_{s^*1})$	$(l^*_{s^*2}, u^*_{s^*2})$	$(l^*_{s^*3}, u^*_{s^*3})$	$(l^*_{s^*4}, u^*_{s^*4})$	$(l^*_{s^*5}, u^*_{s^*5})$
$(0,0)$	$(0,5)$	$(0,5)$	$(0,3)$	$(0,10)$

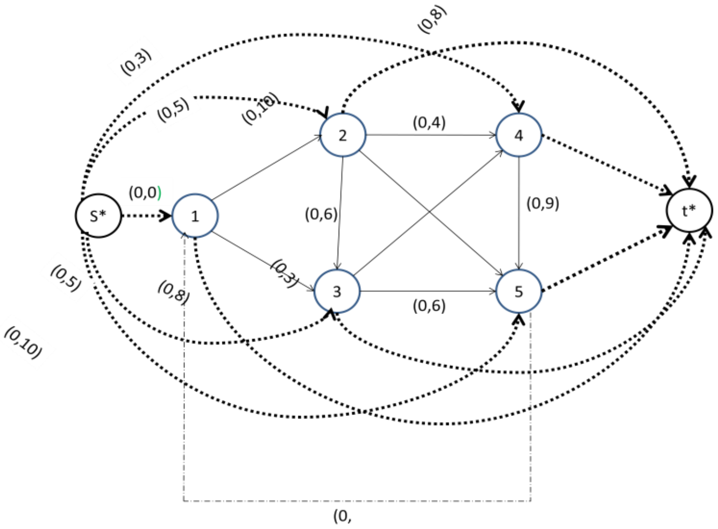


Figure 2.7

Using step 4(d), find the value of $u^*_{it^*}$, $i \in N$ are given in Table 2.5 and shown in Figure 2.8

Table 2.5

$i=1$	$i=2$	$i=3$	$i=4$	$i=5$
$u^*_{1t^*} = \sum_{\substack{r\in(1,2,3,4,5)\\(1,r)\in A}} l_{1r}$	$u^*_{2t^*} = \sum_{\substack{r\in(1,2,3,4,5)\\(2,r)\in A}} l_{2r}$	$u^*_{3t^*} = \sum_{\substack{r\in(1,2,3,4,5)\\(3,r)\in A}} l_{3r}$	$u^*_{4t^*} = \sum_{\substack{r\in(1,2,3,4,5)\\(4,r)\in A}} l_{4r}$	$u^*_{5t^*} = \sum_{\substack{r\in(1,2,3,4,5)\\(5,r)\in A}} l_{5r}$
$u^*_{1t^*} = l_{11}+l_{12}+$ $l_{13}+l_{14}+l_{15}$	$u^*_{2t^*} = l_{21}+l_{22}+$ $l_{23}+l_{24}+l_{25}$	$u^*_{3t^*} = l_{31}+l_{32}+$ $l_{33}+l_{34}+l_{35}$	$u^*_{4t^*} = l_{41}+l_{42}+$ $l_{43}+l_{44}+l_{45}$	$u^*_{5t^*} = l_{51}+l_{52}+$ $l_{53}+l_{54}+l_{55}$
$(1,2),(1,3) \in A$	$(2,,3),(2,4),(2,5) \in A$	$(3,4),(3,5) \in A$	Only $(4,5) \in A$	Only $(5,1) \in A$
$u^*_{1t^*} = l_{12}+l_{13}$	$u^*_{2t^*} = l_{23}+l_{24}+l_{25}$	$u^*_{3t^*} = l_{34}+l_{35}$	$u^*_{4t^*} = l_{45}$	$u^*_{5t^*} = l_{51}$
$u^*_{1t^*} = 5+3$	$u^*_{2t^*} = 2+1+3$	$u^*_{3t^*} = 2+4$		
$u^*_{1t^*} = 8$	$u^*_{2t^*} = 6$	$u^*_{3t^*} = 6$	$u^*_{4t^*} = 3$	$u^*_{5t^*} = 0$

Table 2.6

$(l^*_{1t^*}, u^*_{1t^*})$	$(l^*_{2t^*}, u^*_{2t^*})$	$(l^*_{3t^*}, u^*_{3t^*})$	$(l^*_{4t^*}, u^*_{4t^*})$	$(l^*_{5t^*}, u^*_{5t^*})$
$(0,8)$	$(0,6)$	$(0,6)$	$(0,3)$	$(0,0)$

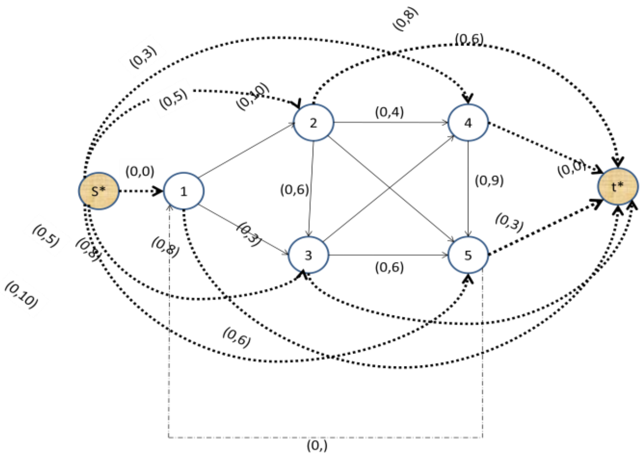


Figure 2.8

Using step 4(e), $u^*_{n1} = \infty$ is shown in Figure 2.9

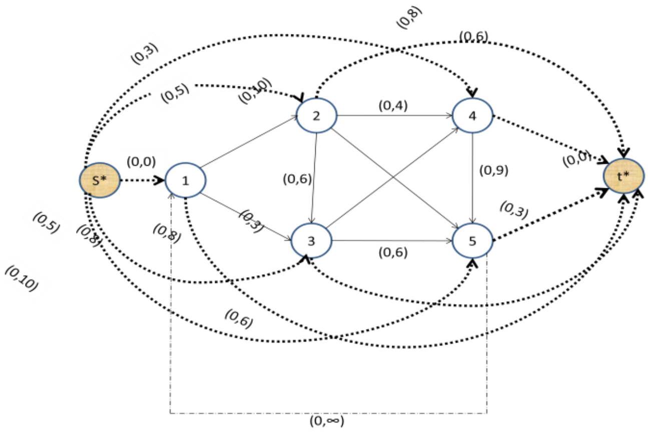


Figure 2.9

Using step 5, find the value of V from Figure 2.10

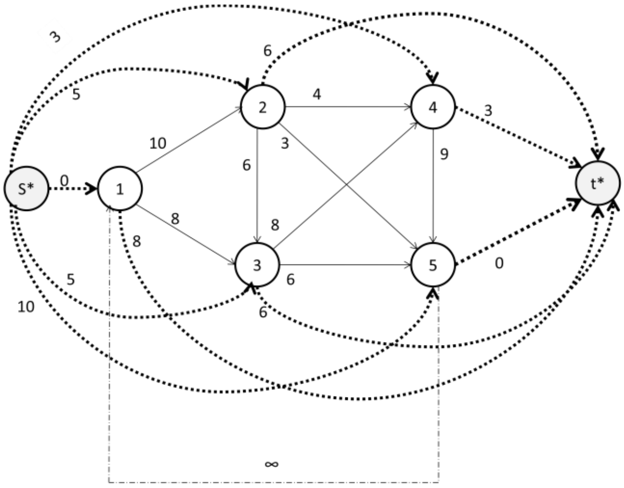


Figure 2.10

Table 2.7

Paths	Maximum flow from source to sink
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3
$s^* - 5 - 1 - 2 - t^*$	1
$s^* - 5 - 1 - 2 - 3 - t^*$	1
	$V = 23$

Table 2.7 shows that $V = \sum_{(i,j)\in A} l_{ij}$, so the network is feasible.

Using step 6, find the critical capacity and modified lower bound of each arc of given network G^* given in Figure 2.10. and the procedure has been shown from Figure 2.11(a) to Figure 2.18(b) and in Table 2.8(a) to 2.8(b) for each arc.

Arc (1,2)

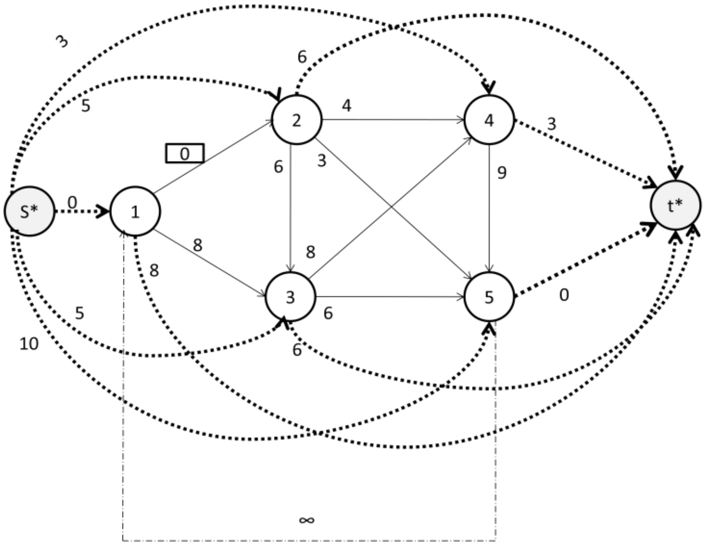


Figure 2.11(a)

When the capacity of (1,2) arc is zero shown in Figure 2.11(a) then maximum flow from source to sink is determined by following paths which are given in Table 2.8(a)

Table 2.8(a)

Paths	Flow
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3
$s^* - 5 - 1 - 3 - t^*$	1
	$v(0) = 22$

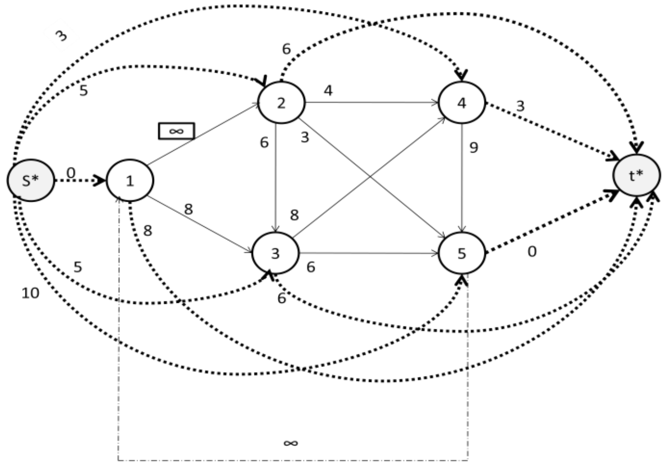


Figure 2.11(b)

When the capacity of (1,2) arc is infinity shown in Figure 2.11(b) then maximum flow from source to sink is determined by following paths which are given in Table 2.8(b)

Table 2.8(b)

Paths	Flow
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3
$s^* - 5 - 1 - 2 - 3 - t^*$	1
$s^* - 5 - 1 - 2 - t^*$	1
	$v(\infty) = 23$

Then k^{*}_{12} (critical capacity of arc $(1,2)$ with respect to G^{*}) = $v(\infty)-v(0)=23-22=1$

Hence $\hat{l}_{12}$ (modified lower bound on the flow on (1,2) in G) $=l_{12}+k^{*}_{12}=5+1=6$.

Arc (1,3)

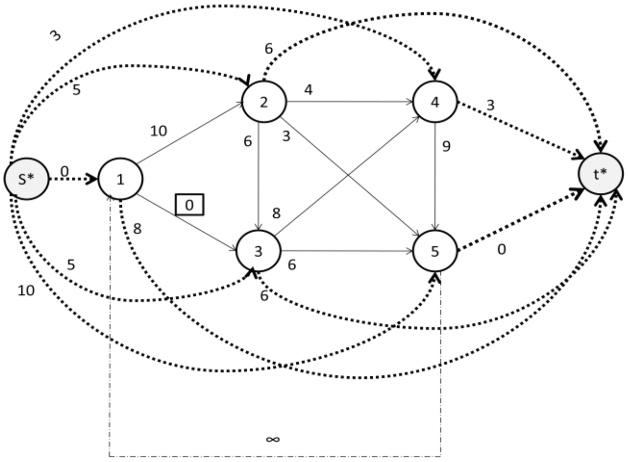


Figure 2.12(a)

When the capacity of (1,3) arc is zero then maximum flow from source to sink is determined by following paths which are given in Table 2.9(a)

Table 2.9(a)

Paths	Flow
$s^{*}-5-1-t^{*}$	8
$s^{*}-2-t^{*}$	5
$s^{*}-3-t^{*}$	5
$s^{*}-4-t^{*}$	3
$s^{*}-5-1-2-3-t^{*}$	1
$s^{*}-5-1-2-t^{*}$	1
	$v(0)=23$

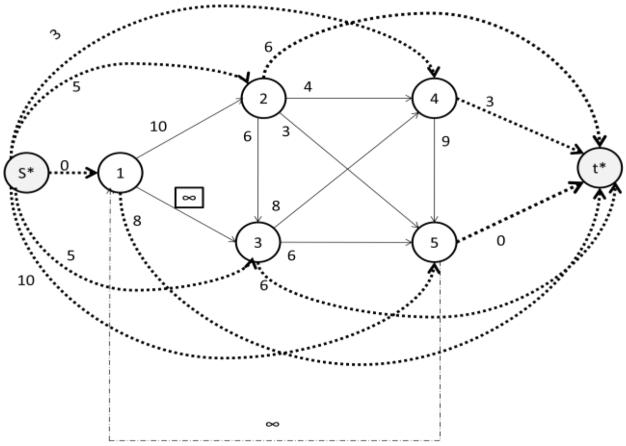


Figure 2.12(b)

When the capacity of (1,3) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.9(b)

Table 2.9(b)

Paths	Flow
$s^*-5-1-t^*$	8
s^*-2-t^*	5
s^*-3-t^*	5
s^*-4-t^*	3
$s^*-5-1-3-t^*$	1
$s^*-5-1-2-t^*$	1
	$v(\infty)=23$

Then k^*_{13} (critical capacity of arc $(1,3)$ with respect to G^*) = $v(\infty)-v(0)=23-23=0$

Hence $\hat{l}_{13}$ (modified lower bound on the flow on (1,3) in G) $=l_{13}+k^*_{13}=l_{13}+0=l_{13}$.

Arc (2,3)

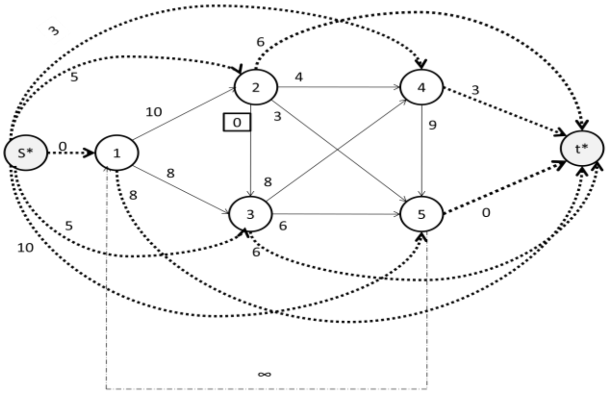


Figure 2.13(a)

When the capacity of (2,3) arc is zero then maximum flow from source to sink is determined by following paths which are given in Table 2.10(a)

Table 2.10(a)

Paths	Flow
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3
$s^* - 5 - 1 - 2 - t^*$	1
$s^* - 5 - 1 - 3 - t^*$	1
	$v(\infty) = 23$

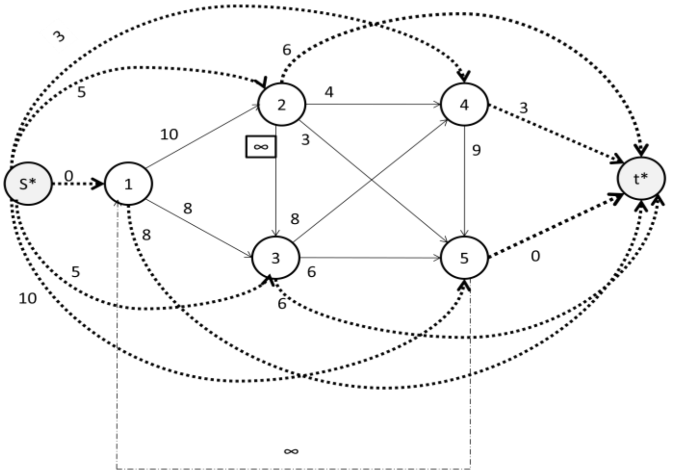


Figure 2.13(b)

When the capacity of (2,3) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.10(b)

Table 2.10(b)

Paths	Flow
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3

$s^*-5-1-2-3-t^*$	1
$s^*-5-1-2-t^*$	1
	$v(\infty)=23$

Then k^*_{23} (critical capacity of arc $(2,3)$ with respect to G^*) = $v(\infty)-v(0)=23-23=0$

Hence $\hat{l}_{23}$ (modified lower bound on the flow on (2,3) in G) $=l_{23}+k^*_{23}=l_{23}+0=l_{23}$.

Arc (2,4)

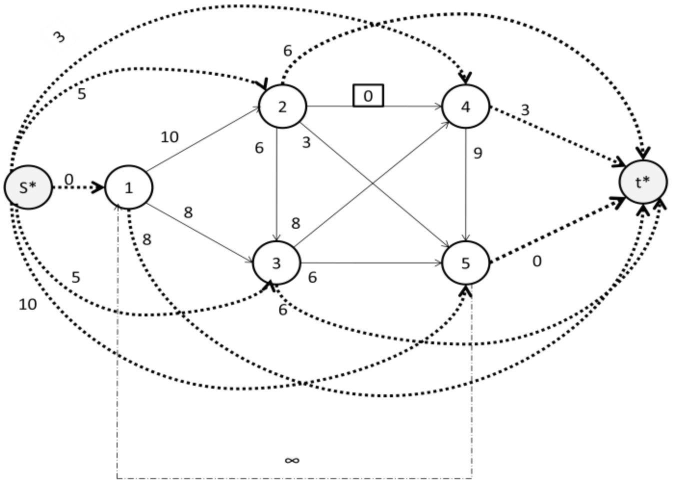


Figure 2.14(a)

When the capacity of (2,4) arc is zero then maximum flow from source to sink is determinedby following paths which are given in Table 2.11(a)

Table 2.11(a)

Paths	Flow
$s^*-5-1-t^*$	8
s^*-2-t^*	5
s^*-3-t^*	5
s^*-4-t^*	3
$s^*-5-1-2-3-t^*$	1
$s^*-5-1-2-t^*$	1
	$v(0)=23$

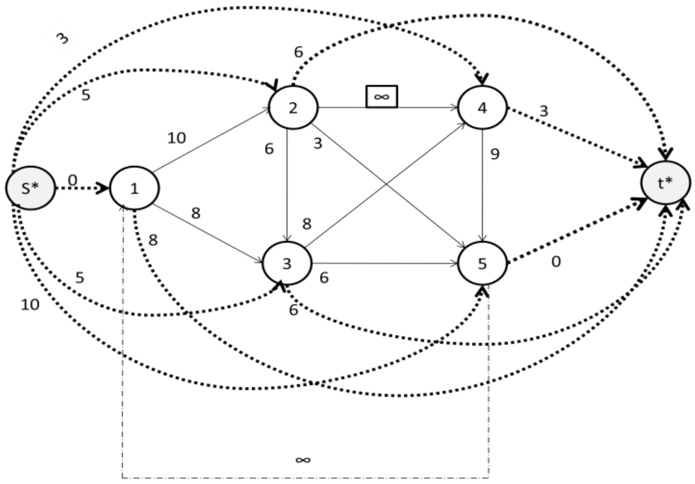


Figure 2.14(b)

When the capacity of (2,4) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.11(b)

Table 2.11(b)

Paths	Flow
$s^*-5-1-t^*$	8
s^*-2-t^*	5
s^*-3-t^*	5
s^*-4-t^*	3
$s^*-5-1-2-3-t^*$	1
$s^*-5-1-2-t^*$	1
	$v(\infty)=23$

Then k^*_{24} (critical capacity of arc $(2,4)$ with respect to G^*) = $v(\infty)-v(0)=23-23=0$

Hence $\hat{l}_{24}$ (modified lower bound on the flow on (2,4) in G) $=l_{24}+k^*_{24}=l_{24}+0=l_{24}$.

Arc (2,5)

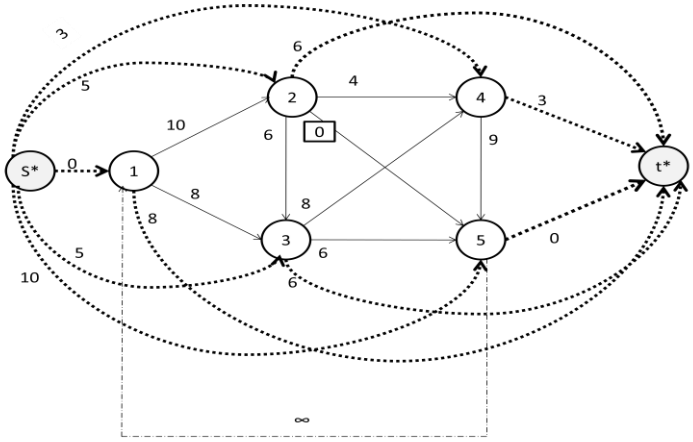


Figure 2.15(a)

When the capacity of (2,5) arc is zero then maximum flow from source to sink is determined by following paths which are given in Table 2.12(a)

Table 2.12(a)

Paths	Flow
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3
$s^* - 5 - 1 - 2 - 3 - t^*$	1
$s^* - 5 - 1 - 2 - t^*$	1
	$v(0) = 23$

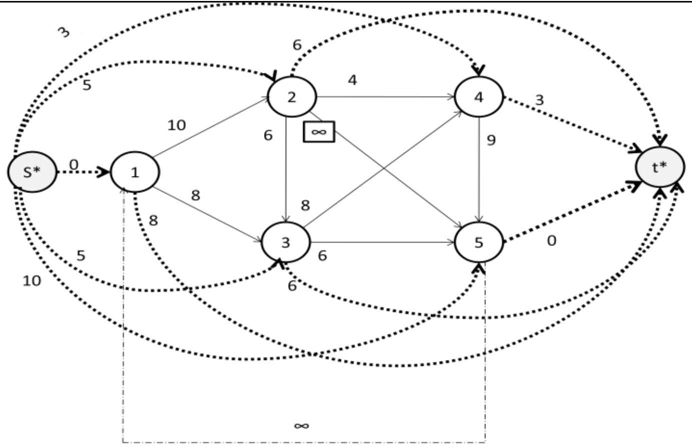


Figure 2.15(b)

When the capacity of (2,5) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.12(b)

Table 2.12(b)

Paths	Flow
$s^*-5-1-t^*$	8
s^*-2-t^*	5
s^*-3-t^*	5
s^*-4-t^*	3
$s^*-5-1-2-3-t^*$	1
$s^*-5-1-2-t^*$	1
	$v(\infty)=23$

Then k^*_{25} (critical capacity of arc $(2,5)$ with respect to G^*) = $v(\infty)-v(0)=23-23=0$

Hence $\hat{l}_{25}$ (modified lower bound on the flow on (2,5) in G) $=l_{25}+k^*_{25}=l_{25}+0=l_{25}$.

Arc (3,4)

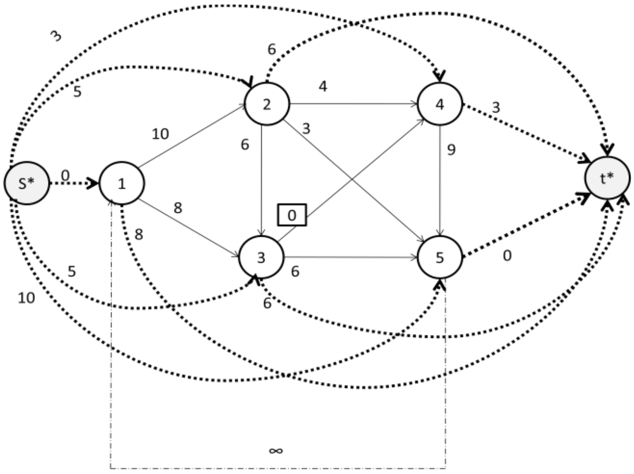


Figure 2.16(a)

When the capacity of (3,4) arc is zero then maximum flow from source to sink is determined by following paths which are given in Table 2.13(a)

Table 2.13(a)

Paths	Flow
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3
$s^* - 5 - 1 - 2 - 3 - t^*$	1
$s^* - 5 - 1 - 2 - t^*$	1
	$v(0) = 23$

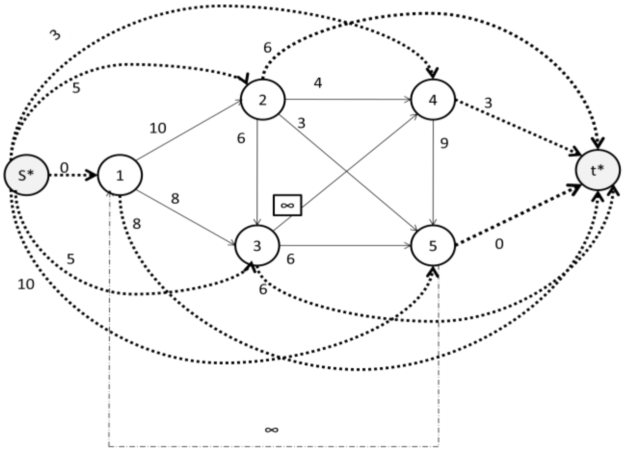


Figure 2.16(b)

When the capacity of (3,4) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.13(b)

Table 2.13(b)

Paths	Flow
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3
$s^* - 5 - 1 - 2 - 3 - t^*$	1
$s^* - 5 - 1 - 2 - t^*$	1
	$v(\infty) = 23$

Then k^*_{34} (critical capacity of arc $(3,4)$ with respect to G^*) = $v(\infty)-v(0)=23-23=0$

Hence $\hat{l}_{34}$ (modified lower bound on the flow on (3,4) in G) $=l_{34}+k^*_{34}=l_{34}+0=l_{34}$.

Arc (3,5)

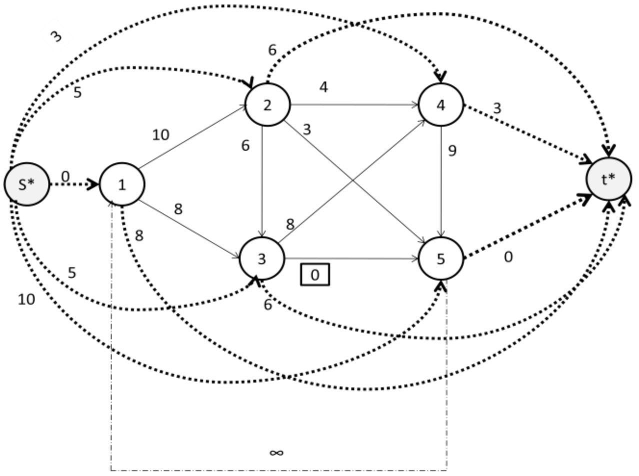


Figure 2.17(a)

When the capacity of (3,5) arc is zero then maximum flow from source to sink is determined by following paths which are given in Table 2.14(a)

Table 2.14(a)

Paths	Flow
$s^*-5-1-t^*$	8
s^*-2-t^*	5
s^*-3-t^*	5
s^*-4-t^*	3
$s^*-5-1-2-3-t^*$	1
$s^*-5-1-2-t^*$	1
	$v(0)=23$

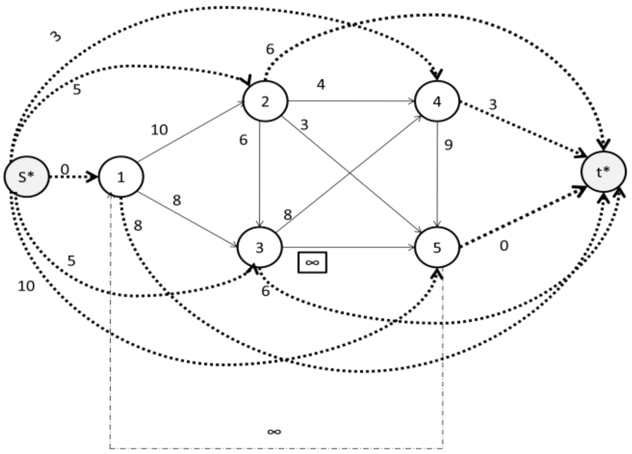


Figure 2.17(b)

When the capacity of (3,5) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.14(b)

Table 2.14(b)

Paths	Flow
$s^* - 5 - 1 - t^*$	8
$s^* - 2 - t^*$	5
$s^* - 3 - t^*$	5
$s^* - 4 - t^*$	3
$s^* - 5 - 1 - 2 - 3 - t^*$	1
$s^* - 5 - 1 - 2 - t^*$	1
	$v(\infty) = 23$

Then k^*_{35} (critical capacity of arc $(3,5)$ with respect to G^*) = $v(\infty) - v(0) = 23 - 23 = 0$

Hence $\hat{l}_{35}$ (modified lower bound on the flow on (3,5) in G) $= l_{35} + k^*_{35} = l_{35} + 0 = l_{35}$.

Arc (4,5)

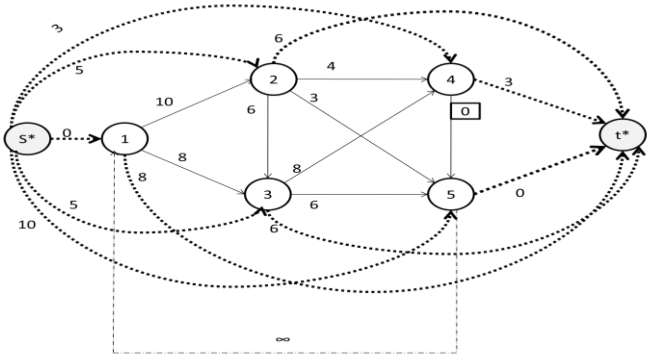


Figure 2.18(a)

When the capacity of (4,5) arc is zero then maximum flow from source to sink is determined by following paths which are given in Table 2.15(a)

Table 2.15(a)

Paths	Flow
$s^*-5-1-t^*$	8
s^*-2-t^*	5
s^*-3-t^*	5
s^*-4-t^*	3
$s^*-5-1-2-3-t^*$	1
$s^*-5-1-2-t^*$	1
	$v(0)=23$

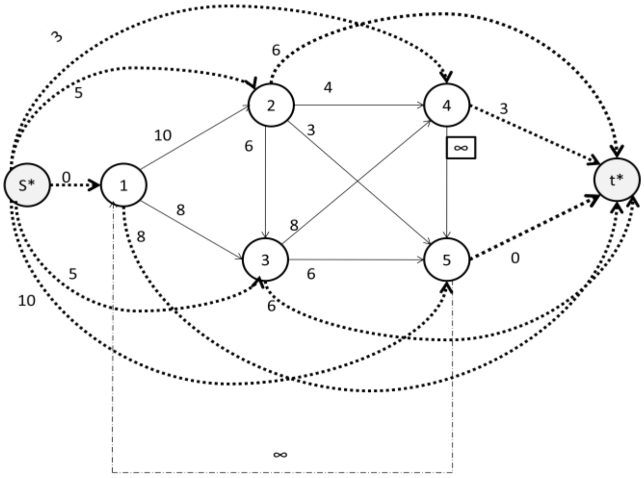


Figure 2.18(b)

When the capacity of (4,5) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.15(b)

Table 2.15(b)

Paths	Flow
$s^*-5-1-t^*$	8
s^*-2-t^*	5
s^*-3-t^*	5
s^*-4-t^*	3
$s^*-5-1-2-3-t^*$	1

$s^*-5-1-2-t^*$	1
	$v(\infty)=23$

Then k^*_{45} (critical capacity of arc $(4,5)$ with respect to G^*) = $v(\infty)-v(0)=23-23=0$

Hence $\hat{l}_{45}$ (modified lower bound on the flow on (4,5) in G) $=l_{45}+k^*_{45}=l_{45}+0=l_{45}$.

Using Step 8 the flow on enlarged network is shown in the Figure 2.19 in small square on each arc

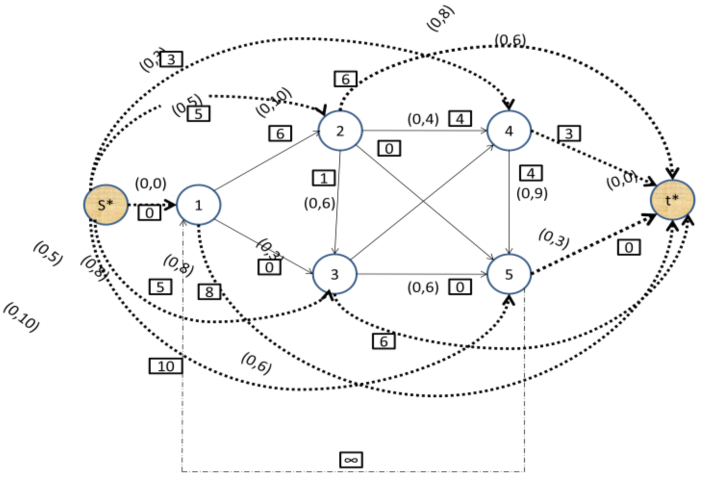
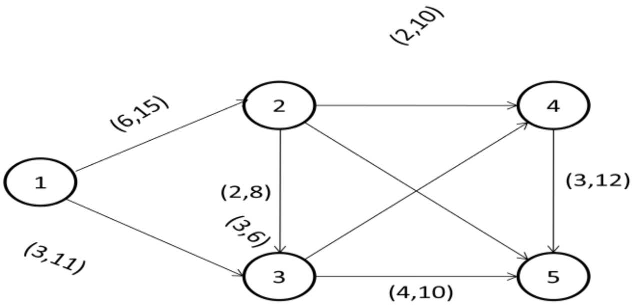


Figure 2.19

The amount of flow on each arc in a flow vector is marked inside a small square by the side of the arc, verify that f is feasible [e.g., total amount reaching 5 is 14 units, 4 units along (4,5) and 10 units along $(s^*,5)$.Total amount leaving 5 is 14 units along (5,1).So conservation holds at 5,etc.]. This flow vector is 23.

Using Step 9, the modified network G is given in Figure 2.20, the critical capacity of each arc of modified network using step 10 has been shown from Figure 2.21(a) to Figure 2.28(b) and Table 2.16 to Table 2.23 and current capacity of each arc using step 11.To check which arc is forward arc or not the step 12 is used.



Modified network (G) Figure 2.20

Arc (1,2)

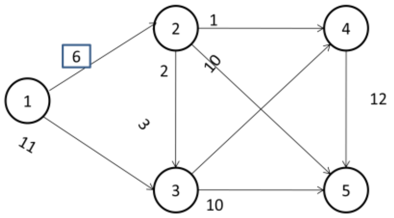


Figure 2.21(a)

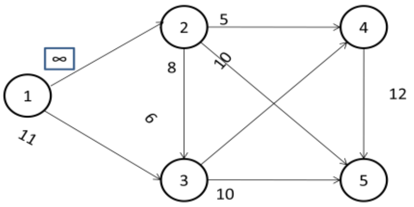


Figure 2.21(b)

When the capacity of (1,2) arc is $\hat{l}_{12}$ then maximum flow from source to sink and when the capacity of (1,2) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.16.

Table 2.16

Paths	Flow	Paths	Flow
$1-3-5$	10	$1-2-3-5$	8
$1-2-5$	3	$1-2-5$	6
$1-2-3-4-5$	2	$1-2-4-5$	5
$1-2-4-5$	1	$1-3-4-5$	7
$1-3-4-5$	1	$1-3-5$	2
	$v(\hat{l}_{12}) = 17$		$v(\infty) = 28$

$\hat{k}_{12}$ (Critical capacity of arc $(1,2)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{12})$

$\hat{k}_{12} = 28 - 17 = 11$

k_{12} (Current capacity of arc $(1,2)$ with respect to modified network G) $= 15 - 6 = 9$

Then $\hat{k}_{12} > k_{12}$

Therefore, arc (1,2) is a forward arc.

Arc (1,3)

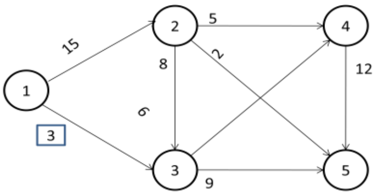


Figure 2.22(a)

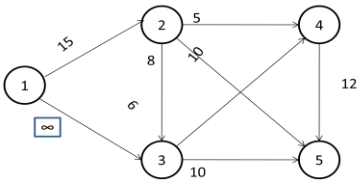


Figure 2.22(b)

When the capacity of (1,3) arc is $\hat{l}_{13}$ then maximum flow from source to sink and when the capacity of (1,3) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.17.

Table 2.17

Paths	Flow	Paths	Flow
$1-2-3-5$	8	$1-3-5$	10
$1-2-5$	6	$1-3-4-5$	10
$1-3-4-5$	2	$1-2-5$	6
$1-2-4-5$	1	$1-2-4-5$	2
$1-3-5$	1		
	$v(\hat{l}_{13}) = 18$		$v(\infty) = 28$

$\hat{k}_{13}$ (Critical capacity of arc $(1,3)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{13})$

$\hat{k}_{13} = 28 - 18 = 10$

k_{13} (Current capacity of arc $(1,3)$ with respect to modified network G) $= 11 - 3 = 8$

Then $\hat{k}_{13} > k_{13}$

Therefore, arc (1,3) is a forward arc.

Arc (2,3)

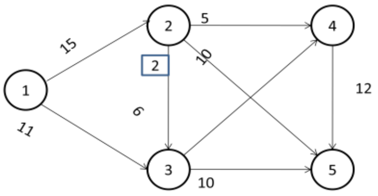


Figure 2.23(a)

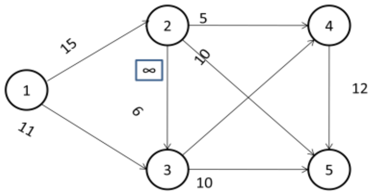


Figure 2.23(b)

When the capacity of (2,3) arc is $\hat{l}_{23}$ then maximum flow from source to sink and when the capacity of (2,3) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.18.

Table 2.18

Paths	Flow	Paths	Flow
$1-2-5$	6	$1-2-5$	6
$1-3-5$	10	$1-3-5$	10
$1-2-4-5$	5	$1-2-3-4-5$	9
$1-2-3-4-5$	2	$1-3-4-5$	1
$1-3-4-5$	1		
	$v(\hat{l}_{23})=24$		$v(\infty)=26$

$\hat{k}_{23}$ (Critical capacity of arc $(2,3)$ with respect to modified network G) $=v(\infty)-v(\hat{l}_{23})$

$\hat{k}_{23}=26-24=2$

k_{23} (Current capacity of arc $(2,3)$ with respect to modified network G) $=8-2=6$

$\hat{k}_{23}<k_{23}$

Therefore, arc $(2,3)$ is not a forward arc.

Arc (2,4)

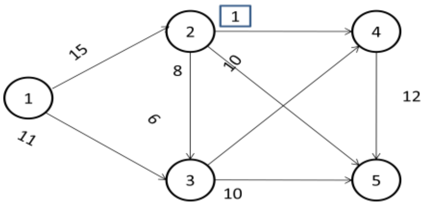


Figure 2.24(a)

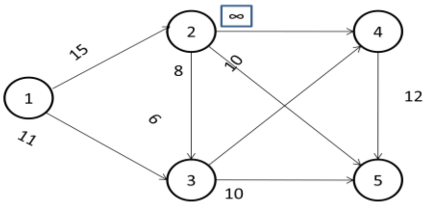


Figure 2.24(b)

When the capacity of (2,4) arc is $\hat{l}_{24}$ then maximum flow from source to sink and when the capacity of (2,4) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.19.

Table 2.19

Paths	Flow	Paths	Flow
$1-2-3-5$	8	$1-2-3-5$	8
$1-3-4-5$	10	$1-3-4-5$	10
$1-2-5$	6	$1-2-5$	6
$1-2-4-5$	1	$1-2-4-5$	1
$1-3-5$	1	$1-3-5$	1
	$v(\hat{l}_{24}) = 26$		$v(\infty) = 26$

$\hat{k}_{24}$ (Critical capacity of arc $(2,4)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{24})$

$\hat{k}_{24} = 26 - 26 = 0$

k_{24} (Current capacity of arc $(2,4)$ with respect to modified network G) $= 5 - 1 = 4$

Then $\hat{k}_{24} < k_{24}$

Therefore, arc (2,4) is a not a forward arc.

Arc (2,5)

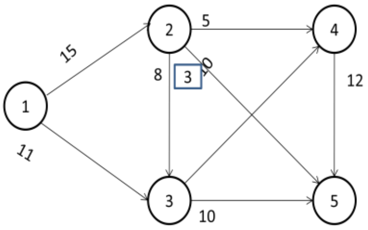


Figure 2.25(a)

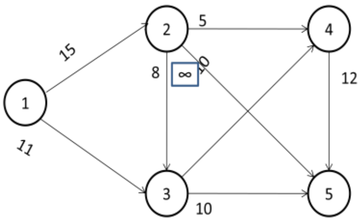


Figure 2.25(b)

When the capacity of (2,5) arc is $\hat{l}_{25}$ then maximum flow from source to sink and when the capacity of (2,5) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.20.

Table 2.20

Paths	Flow	Paths	Flow
$1-3-5$	10	$1-2-5$	15
$1-2-3-4-5$	8	$1-3-5$	10
$1-2-4-5$	4	$1-3-4-5$	1
$1-2-5$	3		
	$v(\hat{l}_{25})=25$		$v(\infty)=26$

$\hat{k}_{25}$ (Critical capacity of arc $(2,5)$ with respect to modified network G) $=v(\infty)-v(\hat{l}_{25})$

$\hat{k}_{25}=26-25=1$

k_{25} (Current capacity of arc $(2,5)$ with respect to modified network G) $=6-3=3$

Then $\hat{k}_{25}<k_{25}$

Therefore, arc $(2,5)$ is not a forward arc.

Arc (3,4)

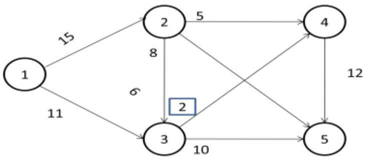


Figure 2.26(a)

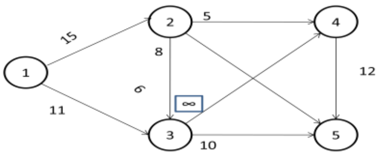


Figure 2.26(b)

When the capacity of (3,4) arc is $\hat{l}_{34}$ then maximum flow from source to sink and when the capacity of (3,4) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.21.

Table 2.21

Paths	Flow	Paths	Flow
$1-3-5$	10	$1-3-4-5$	11
$1-2-3-4-5$	2	$1-2-3-5$	8
$1-2-5$	6	$1-2-5$	6
$1-2-4-5$	5	$1-2-4-5$	1
	$v(\hat{l}_{34})=23$		$v(\infty)=26$

$\hat{k}_{34}$ (Critical capacity of arc $(3,4)$ with respect to modified network G) $=v(\infty)-v(\hat{l}_{34})$

$\hat{k}_{34}=26-23=3$

k_{34} (Current capacity of arc $(3,4)$ with respect to modified network G) $=10-2=8$

Then $\hat{k}_{34}<k_{34}$

Therefore, arc (3,4) is a not a forward arc.

Arc (3,5)

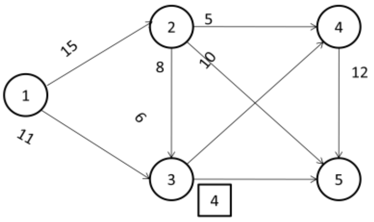


Figure 2.27(a)

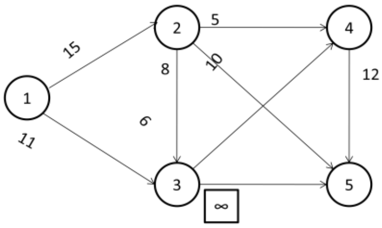


Figure 2.27(b)

When the capacity of (3,5) arc is $\hat{l}_{35}$ then maximum flow from source to sink and when the capacity of (3,5) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.22.

Table 2.22

Paths	Flow	Paths	Flow
$1-3-4-5$	10	$1-2-3-5$	8
$1-2-3-5$	4	$1-3-5$	11
$1-2-5$	6	$1-2-5$	6
$1-2-4-5$	2	$1-2-4-5$	1
	$v(\hat{l}_{35})=22$		$v(\infty)=26$

$\hat{k}_{35}$ (Critical capacity of arc $(3,5)$ with respect to modified network G) $=v(\infty)-v(\hat{l}_{35})$

$\hat{k}_{35}=26-22=4$

k_{35} (Current capacity of arc $(3,5)$ with respect to modified network G) $=10-4=6$

Then $\hat{k}_{35}<k_{35}$

Therefore, arc $(3,5)$ is a not a forward arc.

Arc (4,5)

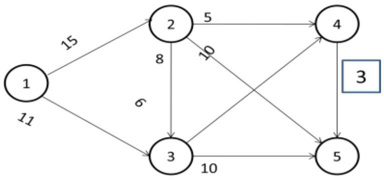


Figure 2.28(a)

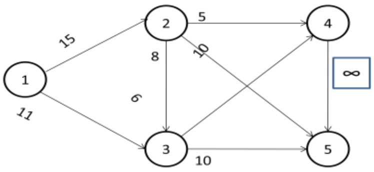


Figure 2.28(b)

When the capacity of (4,5) arc is $\hat{l}_{45}$ then maximum flow from source to sink and When the capacity of (4,5) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.23.

Table 2.23

Paths	Flow	Paths	Flow
$1-2-3-5$	8	$1-2-3-4-5$	8
$1-3-4-5$	3	$1-3-5$	10
$1-3-5$	2	$1-2-5$	6
$1-2-5$	6	$1-2-4-5$	1
		$1-3-4-5$	1
	$v(\hat{l}_{45})=19$		$v(\infty)=26$

$\hat{k}_{45}$ (Critical capacity of arc $(4,5)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{45})$

$\hat{k}_{45} = 26 - 19 = 7$

k_{45} (Current capacity of arc $(4,5)$ with respect to modified network G) $= 12 - 3 = 9$

Then $\hat{k}_{45} < k_{45}$

Therefore, arc $(4,5)$ is not a forward arc.

Using step 13, find the maximum flow in modified network G

For $\lambda \in [0,4]$ we get various max flow vectors all of value 26.There may be more max flow vectors for e.g. $(f_{12}, f_{13}, f_{23}, f_{24}, f_{25}, f_{34}, f_{35}, f_{45})$ is:

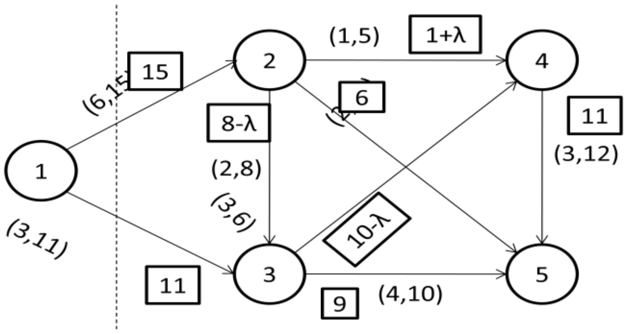


Figure 2.29

The amount of flow on each arc in a flow vector is marked inside a small square by the side of the arc.

Total amount reaching 2 is 15 units and total amount of leaving at 2 node is 6 from (2,5), $1+\lambda$ from (2,4) and $8-\lambda$ from (2,3).Total flow is $6+1+\lambda+8-\lambda$ =15.

Total amount reaching 3 is $19-\lambda$ units and total amount of leaving at 3 node is 9 from (3,5), $10-\lambda$ .Total flow is $9+10-\lambda$ $=19-\lambda$.

Example 2: Consider the feasible directed network G shown in Figure 2.30.

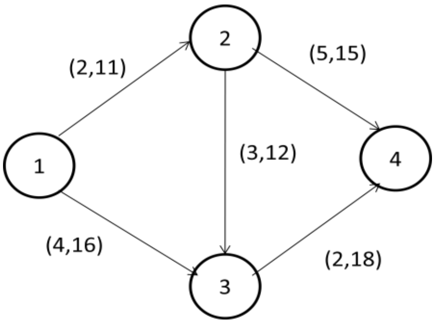


Figure 2.30

Introduce the arc (4,1) with the lower bound on the flow is zero and upper bound as infinity. Using step 2 to step 5, feasibility conditions is checked and the procedure is given from Figure 2.31 to Figure 2.36

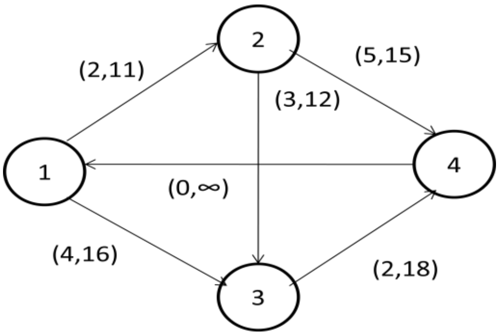


Figure 2.31

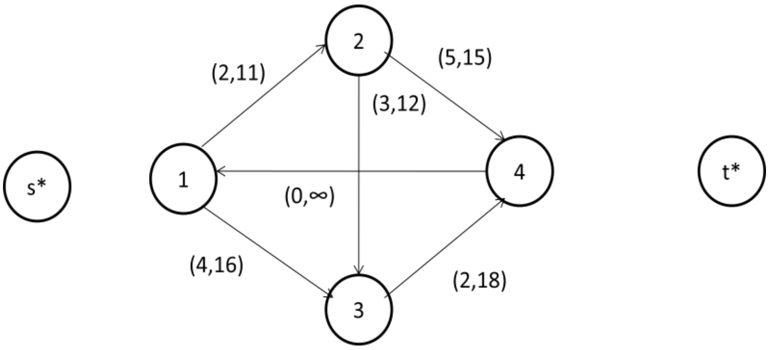


Figure 2.32

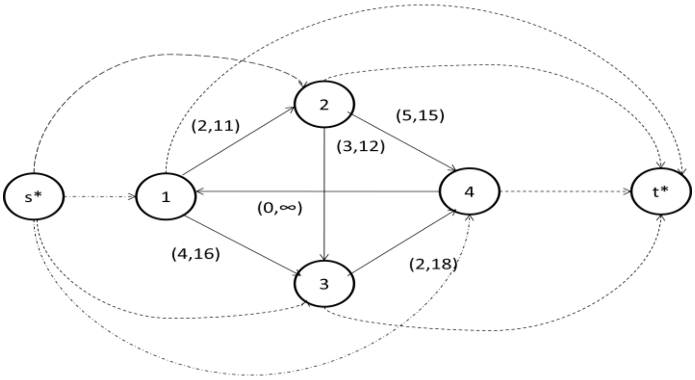


Figure 2.33

Using step 4(b), find the value of $u^*_{ij}, \forall (i,j) \in A$ are given in Table 2.24 and shown in Figure 2.34

Table 2.24

Take $(1,2)$ arc	Take $(1,3)$ arc	Take $(2,3)$ arc	Take $(2,4)$ arc	Take $(3,4)$ arc
$u^*_{12} = u_{12} - l_{12}$	$u^*_{13} = u_{13} - l_{13}$	$u^*_{13} = u_{13} - l_{13}$	$u^*_{24} = u_{24} - l_{24}$	$u^*_{34} = u_{34} - l_{34}$
$u^*_{12} = 11 - 2$	$u^*_{13} = 16 - 4$	$u^*_{13} = 12 - 3$	$u^*_{24} = 15 - 5$	$u^*_{34} = 18 - 2$
$u^*_{12} = 9$	$u^*_{13} = 12$	$u^*_{13} = 9$	$u^*_{24} = 10$	$u^*_{34} = 16$

Table 2.25

(l^*_{12}, u^*_{12})	(l^*_{13}, u^*_{13})	(l^*_{23}, u^*_{23})	(l^*_{24}, u^*_{24})	(l^*_{34}, u^*_{34})
(0,9)	(0,12)	(0,9)	(0,10)	(0,16)

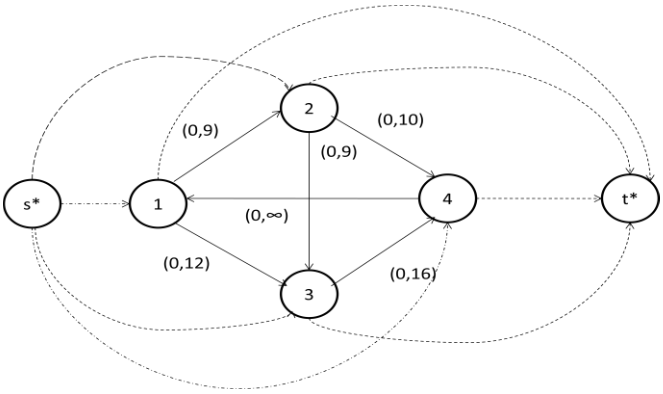


Figure 2.34

Using step 4(c), find the value of $u^*_{s^*i}, i \in N$ are given in Table 2.26 and shown in Figure 2.35

Table 2.26

$i=1$	$i=2$	$i=3$	$i=4$
$u^*_{s^*1} = \sum_{\substack{r\in(1,2,3,4)\\(r,1)\in A}} l_{r1}$	$u^*_{s^*2} = \sum_{\substack{r\in(1,2,3,4)\\(r,2)\in A}} l_{r2}$	$u^*_{s^*3} = \sum_{\substack{r\in(1,2,3,4)\\(r,3)\in A}} l_{r3}$	$u^*_{s^*4} = \sum_{\substack{r\in(1,2,3,4)\\(r,4)\in A}} l_{r4}$
$u^*_{s^*1} = l_{11} + l_{21} + l_{31} + l_{41}$	$u^*_{s^*2} = l_{12} + l_{22} + l_{32} + l_{42}$	$u^*_{s^*3} = l_{13} + l_{23} + l_{33} + l_{43}$	$u^*_{s^*4} = l_{14} + l_{24} + l_{34} + l_{44}$
$\text{only}\,(4,1) \in A$	$\text{only}\,(1,2) \in A$	$(1,3),(2,3) \in A$	$(2,4),(3,4) \in A$
$u^*_{s^*1} = l_{41}$	$u^*_{s^*2} = l_{12}$	$u^*_{s^*3} = l_{13} + l_{23}$	$u^*_{s^*4} = l_{24} + l_{34}$
		$u^*_{s^*3} = 4+3$	$u^*_{s^*4} = 5+2$
$u^*_{s^*1} = 0$	$u^*_{s^*2} = 2$	$u^*_{s^*3} = 7$	$u^*_{s^*4} = 7$

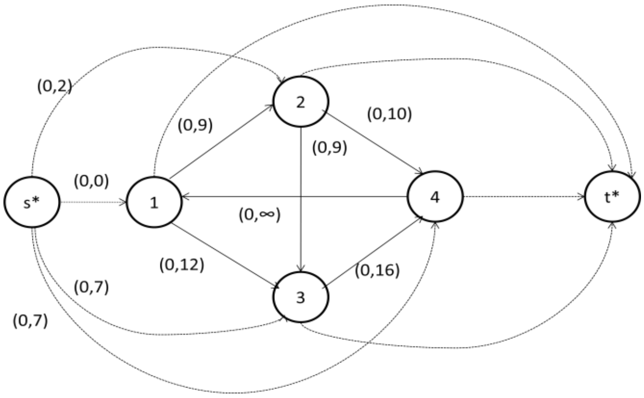


Figure 2.35

Using step 4(d), find the value of $u^*_{it^*}, i \in N$ are given in Table 2.27 and shown in Figure 2.36

Table 2.27

$i=1$	$i=2$	$i=3$	$i=4$
$u^*_{1t^*}=\sum_{\substack{r\in(1,2,3,4)\\(1,r)\in A}} l_{1r}$	$u^*_{2t^*}=\sum_{\substack{r\in(1,2,3,4)\\(2,r)\in A}} l_{2r}$	$u^*_{3t^*}=\sum_{\substack{r\in(1,2,3,4)\\(3,r)\in A}} l_{3r}$	$u^*_{4t^*}=\sum_{\substack{r\in(1,2,3,4)\\(4,r)\in A}} l_{4r}$
$u^*_{1t^*}=l_{11}+l_{12}+l_{13}+l_{14}$	$u^*_{2t^*}=l_{21}+l_{22}+l_{23}+l_{24}$	$u^*_{3t^*}=l_{31}+l_{32}+l_{33}+l_{34}$	$u^*_{4t^*}=l_{41}+l_{42}+l_{43}+l_{44}$
only $(1,2),(1,3)\in A$	only $(2,3),(2,4)\in A$	only $(3,4)\in A$	only $(4,1)\in A$
$u^*_{1t^*}=l_{12}+l_{13}$	$u^*_{2t^*}=l_{23}+l_{24}$	$u^*_{3t^*}=l_{34}$	$u^*_{4t^*}=l_{41}$
$u^*_{1t^*}=2+4$	$u^*_{2t^*}=3+5$		
$u^*_{1t^*}=6$	$u^*_{2t^*}=8$	$u^*_{3t^*}=2$	$u^*_{4t^*}=0$

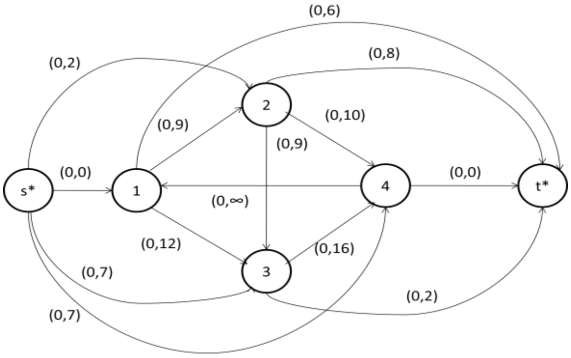


Figure 2.36

Using step 5,find the value of V

Table 2.28

Paths	Value of V
s^*-3-t^*	2
$s^*-4-1-2-t^*$	7
$s^*-3-4-1-t^*$	5
s^*-2-t^*	1
$s^*-2-4-1-t^*$	1
	16

Table 2.28 shows that $V = \sum_{(i,j)\in A} l_{ij}$, so the network is feasible.

Using step 6, find the critical capacity and modified lower bound of each arc of given network G^* given in Figure 2.36. and the procedure has been shown from Figure 2.37(a) to Figure 2.37(b) and in Table 2.29(a) to 2.29(b) for each arc.

Arc (1,2)

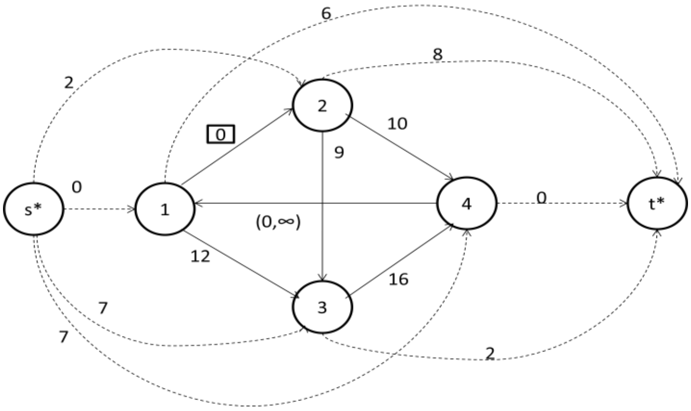


Figure 2.37(a)

When the capacity of (1,2) arc is zero then maximum flow from source to sink is determined by following paths which are given in Table 2.29(a).

Table 2.29(a)

Paths	Flow
$s^* - 3 - t^*$	2
$s^* - 4 - 1 - t^*$	6
$s^* - 2 - t^*$	2
	$v(0) = 10$

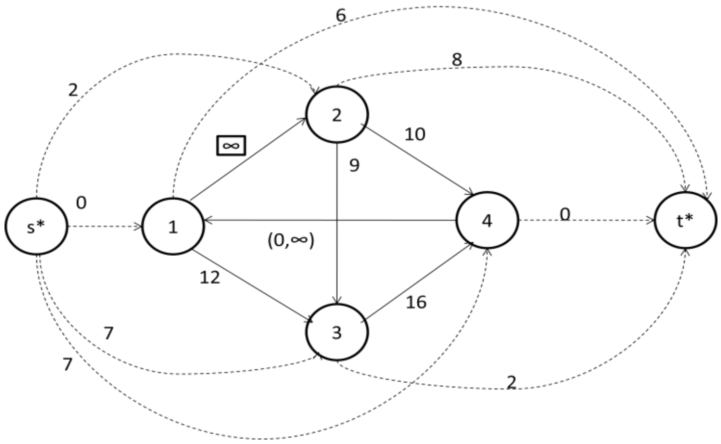


Figure 2.37(b)

When the capacity of (1,2) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.29(b).

Table 2.29(b)

Paths	Flow
s^*-3-t^*	2
$s^*-4-1-2-t^*$	7
$s^*-3-4-1-t^*$	5
s^*-2-t^*	1
$s^*-2-4-1-t^*$	1
	$v(\infty)=16$

Then k^*_{12} (critical capacity of arc $(1,2)$ with respect to G^*) = $v(\infty)-v(0)=16-10=6$

Hence $\hat{l}_{12}$ (modified lower bound on the flow on (1,2) in G) $=l_{12}+k^*_{12}=2+6=8$.

Arc (1,3)

Then k^*_{13} (critical capacity of arc $(1,3)$ with respect to G^*) = $v(\infty)-v(0)=16-16=0$

Hence $\hat{l}_{13}$ (modified lower bound on the flow on (1,3) in G) $=l_{13}+k^*_{13}=l_{13}+0=l_{13}$.

Arc (2,3)

Then k^*_{23} (critical capacity of arc $(2,3)$ with respect to G^*) = $v(\infty)-v(0)=16-16=0$

Hence $\hat{l}_{23}$ (modified lower bound on the flow on (2,3) in G) $=l_{23}+k^*_{23}=l_{23}+0=l_{23}$.

Arc (2,4)

Then k^{*}_{24} (critical capacity of arc $(2,4)$ with respect to G^{*}) = $v(\infty)-v(0)=16-16=0$

Hence $\hat{l}_{24}$ (modified lower bound on the flow on (2,4) in G) $=l_{24}+k^{*}_{24}=l_{24}+0=l_{24}$.

Arc (3,4)

Then k^{*}_{34} (critical capacity of arc $(3,4)$ with respect to G^{*}) = $v(\infty)-v(0)=16-11=0$

Hence $\hat{l}_{34}$ (modified lower bound on the flow on (3,4) in G) $=l_{34}+k^{*}_{34}=2+5=7$.

Using Step 8 the flow on enlarged network is shown in the Figure 2.38 in small square on each arc

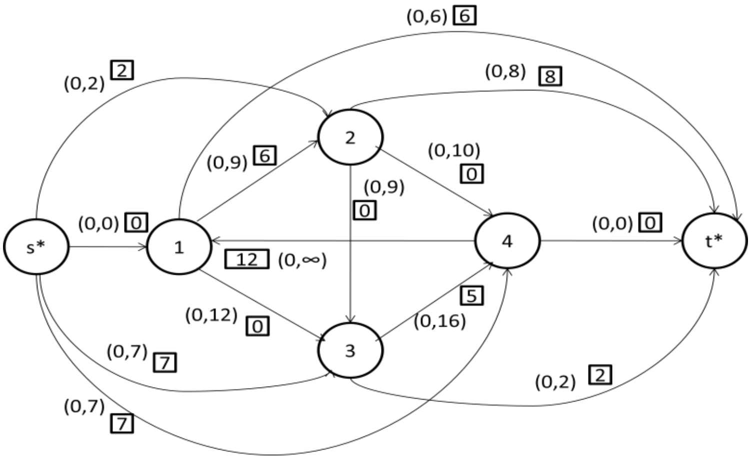
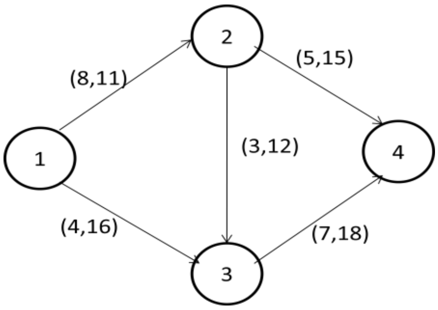


Figure 2.38

The amount of flow on each arc in a flow vector is marked inside a small square by the side of the arc, verify that f is feasible [e.g.,total amount reaching 4 is 12 units, 5 units along (3,4) and 7 units along $(s^{*},4)$ .Total amount leaving 4 is 12 units along (4,1).So conservation holds at 4,etc.].This flow vector is 16.



Modified network (G) Figure 2.39

Using Step 9,the modified network G is given in Figure 2.39, the critical capacity of each arc of modified network using step 10 has been shown from Figure 2.40(a) to Figure 2.44(b) and Table 2.30 to Table 2.34 and current capacity of each arc using step 11.To check which arc is forward arc or not the step 12 is used.

Arc (1,2)

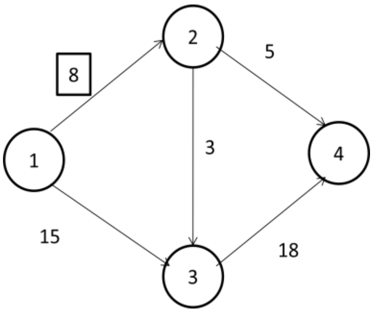


Figure 2.40(a)

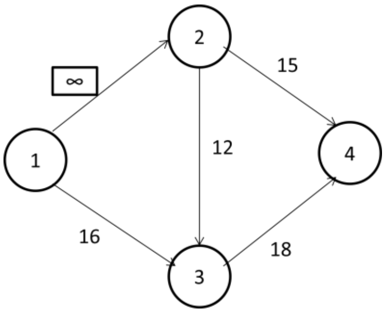


Figure 2.40(b)

When the capacity of (1,2) arc is $\hat{l}_{12}$ then maximum flow from source to sink and When the capacity of (1,2) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.30.

Table 2.30

Paths	Flow	Paths	Flow
$1-3-4$	15	$1-2-4$	15
$1-2-4$	5	$1-2-3-4$	12
$1-2-3-4$	3	$1-3-4$	6
	$v(\hat{l}_{12}) = 23$		$v(\infty) = 33$

$\hat{k}_{12}$ (Critical capacity of arc $(1,2)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{12})$

$\hat{k}_{12} = 33 - 23 = 10$

k_{12} (Current capacity of arc $(1,2)$ with respect to modified network G) $= 11 - 8 = 3$

Then $\hat{k}_{12} > k_{12}$

Therefore , arc (1,2) is a forward arc.

Arc (1,3)

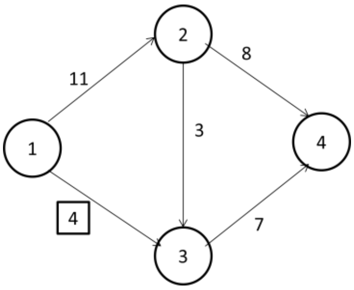


Figure 2.41(a)

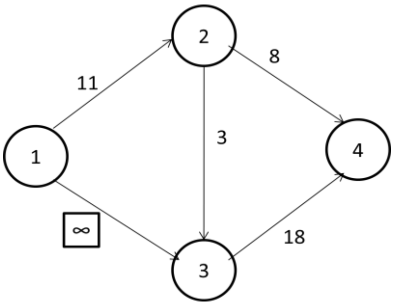


Figure 2.41(b)

When the capacity of (1,3) arc is $\hat{l}_{13}$ then maximum flow from source to sink and When the capacity of (1,3) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.31.

Table 2.31

Paths	Flow	Paths	Flow
$1-2-4$	8	$1-3-5$	18
$1-3-4$	4	$1-2-4$	8
$1-2-3-4$	3		
	$v(\hat{l}_{13}) = 15$		$v(\infty) = 26$

$\hat{k}_{13}$ (Critical capacity of arc $(1,3)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{13})$

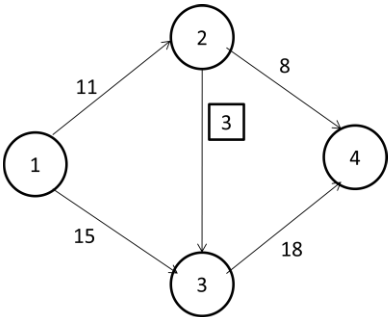
$\hat{k}_{13} = 26 - 15 = 11$

k_{13} (Current capacity of arc $(1,3)$ with respect to modified network G) $= 16 - 4 = 12$

Then $\hat{k}_{13} < k_{13}$

Therefore ,arc (1,3) is not a forward arc.

Arc (2,3)



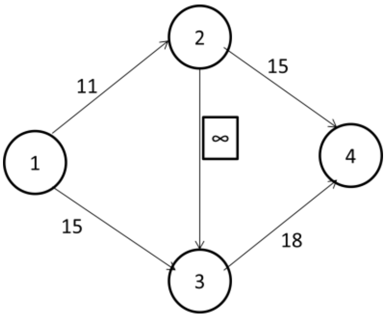


Figure 2.42(a) Figure 2.42(b)

When the capacity of (2,3) arc is $\hat{l}_{23}$ then maximum flow from source to sink and When the capacity of (2,3) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.32.

Table 2.32

Paths	Flow	Paths	Flow
$1-3-4$	15	$1-3-4$	15
$1-2-4$	8	$1-2-3-4$	3
$1-2-3-4$	3	$1-2-4$	8
	$v(\hat{l}_{23})=26$		$v(\infty)=26$

$\hat{k}_{23}$ (Critical capacity of arc $(2,3)$ with respect to modified network G) $=v(\infty)-v(\hat{l}_{23})$

$\hat{k}_{23}=26-26=0$

k_{23} (Current capacity of arc $(2,3)$ with respect to modified network G) $=12-3=9$

$\hat{k}_{23}<k_{23}$

Therefore, arc $(2,3)$ is not a forward arc.

Arc (2,4)

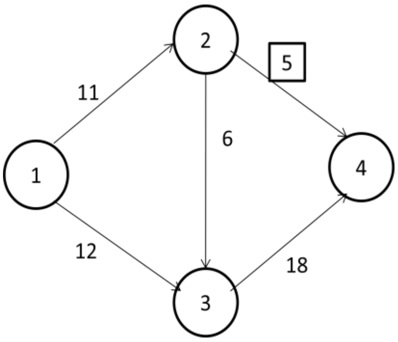


Figure2.43(a)

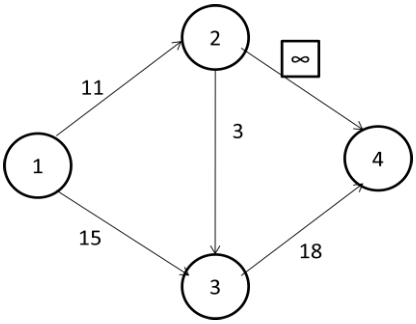


Figure 2.43(b)

When the capacity of (2,4) arc is $\hat{l}_{24}$ then maximum flow from source to sink and When the capacity of (2,4) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.33.

Table 2.33

Paths	Flow	Paths	Flow
$1-3-4$	12	$1-3-4$	15
$1-2-3-4$	6	$1-2-4$	11
$1-2-4$	5		
	$v(\hat{l}_{24})=23$		$v(\infty)=26$

$\hat{k}_{24}$ (Critical capacity of arc $(2,4)$ with respect to modified network G) $=v(\infty)-v(\hat{l}_{24})$

$\hat{k}_{24}=26-23=3$

k_{24} (Current capacity of arc $(2,4)$ with respect to modified network G) $=15-5=10$

Then $\hat{k}_{24}<k_{24}$

Therefore, arc $(2,4)$ is a not a forward arc.

Arc (3,4)

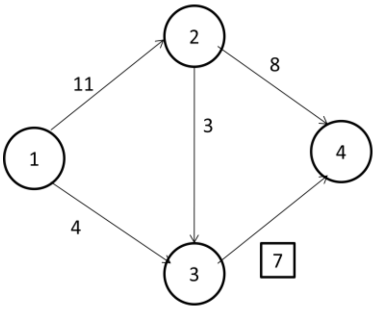


Figure 2.44(a)

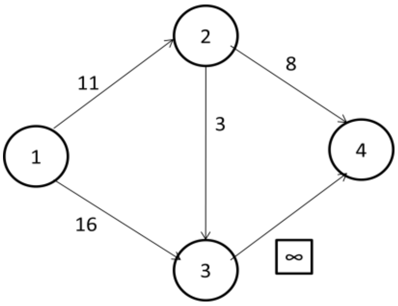


Figure 2.44(b)

When the capacity of (3,4) arc is $\hat{l}_{34}$ then maximum flow from source to sink and When the capacity of (3,4) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 2.34.

Table 2.34

Paths	Flow	Paths	Flow
$1-2-4$	8	$1-3-4$	16
$1-3-4$	4	$1-2-4$	11
$1-2-3-4$	3		
	$v(\hat{l}_{34})=15$		$v(\infty)=27$

$\hat{k}_{34}$ (Critical capacity of arc $(3,4)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{34})$

$\hat{k}_{34} = 27 - 15 = 12$

k_{34} (Current capacity of arc $(3,4)$ with respect to modified network G) $= 18 - 7 = 11$

Then $\hat{k}_{34} > k_{34}$

Therefore, arc $(3,4)$ is a forward arc.

Using step 13, find the maximum flow in modified network G We get various max flow vectors all of value 26.There may be more max flow vectors.

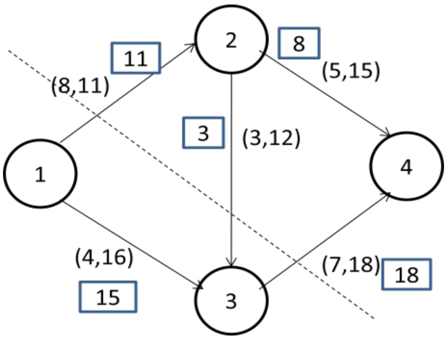


Figure 2.45

Chapter - 3

THE CUT SEARCH ALGORITHM WITH ARC CAPACITIES AND LOWER BOUNDS

3.1 INTRODUCTION

A methodology proposed by Phillips et. al. (1979) to find the cut with arc capacities and lower bounds is reviewed in this chapter.

Consider a flow network $G = (N, A)$ having $n = |N|$ nodes and $m = |A|$ arcs. Source and sink nodes are denoted by s and t respectively. An each arc has lower and upper bound which are denoted by $l(i, j)$ and $u(i, j)$.The flow in an arc (i, j) is denoted by $f(i, j)$ is then restricted by:

$$0 \le l(i, j) \le f(i, j) \le u(i, j) \forall (i, j) \in A$$

The problem of interest here is determining maximal flow, $f^*(s,t)$ of a commodity that could be sent from s to t subject to above restriction and flow conservation constraint for each node except s and t.This conservation constraint can be stated as follows:

$$\sum_{i \in N} f(i, j) = \sum_{k \in N} f(j, k) \text{ each } j \in N \text{ and } a(i, j) \in A \text{ where } j \ne s \, or \, t.$$

Define a set of nodes including s as W and its complimentary set which contains t and other nodes which are not present in W is denoted as $\overline{W}$.

The set of arcs separating W and $\overline{W}$ are denoted as $(W, \overline{W})$ and constitute a cut. The value associated with any cut, $(W, \overline{W})$ is denoted as $K(W, \overline{W})$ and is computed by a difference between $U(W, \overline{W})$ and $L(\overline{W}, W)$. $U(W, \overline{W})$ is the sum of arc capacities on arcs going from W to $\overline{W}$:

$$U(W, \overline{W}) = \sum_{\substack{i \in W, j \in \overline{W} \\ a(i,j) \in A}} u(i, j)$$

and $L(\overline{W},W)$ is the sum of lower bounds on arcs going from $\overline{W}$ to W:

$$L(\overline{W},W) = \sum_{\substack{m\in\overline{W},n\in W \\ a(m,n)\in A}} l(m,n)$$

Then for any $(W,\overline{W})$, the cut value defined as $K(W,\overline{W})$ is given by:

$$K(W,\overline{W}) = U(W,\overline{W}) - L(\overline{W},W).$$

Hence, the minimal cut value, denoted by $K^*(s,t)$, is given by:

$$K^*(s,t) = \min K(W,\overline{W}).$$

$K^*(s,t)$ is the minimal cut value and therefore equal the maximal flow in the network $G=(N,A)$.

The cut search algorithm is designed to locate $K^*(s,t)$, through a two stage procedure. Basically the algorithm consists of establishing three sets identified as T ,W and S , and then expanding these sets node by node until the minimal cut is located. Set T is developed during the backward procedure which is introduced as an accerlation technique.This set is initiated at the sink node and expanded toward the source while computing $K(\overline{T},T)$. If T is expanded to include all nodes except the source then $K(\overline{T},T)$ will define $K^*(s,t)$. If the set T can not be expanded so that $\overline{T}=s$, then a forward procedure is initiated by setting sets W and S equal to the source node. Set W is expanded node by node until set T is reached, that is, until $\overline{W}=t$ and at this point the procedure will terminate. Set S, which is contained in W , is expanded only by adding nodes in W which reduce the cut value as established by the current
$(S,\overline{S})$. That is, only those nodes which will result in a lower $K(S,\overline{S})$ value will be added to S. Therefore, at algorithm termination $K(S,\overline{S})$ will define $K^*(s,t)$.

3.2. SOLUTION PROCEDURE

Determine the $K^*(s,t)$ which is the minimal cut value and equals to the maximal flow in the network $G=(N,A)$.The cut search algorithm is designed to locate $K^*(s,t)$ through a two stage procedure:

1)Backward procedure

2)Forward procedure

Backward procedure:-

The backward procedure starts be defining T as the sink node. $K(\overline{T},T)$ is computed by

$$K(\overline{T},T)=U(\overline{T},T)-L(T,\overline{T}) \qquad ...(3.1)$$

which represents a local minimum. For every set T, a set X' is defined as $X'=(x\,|\,x\in\overline{T}, x\neq s,$ and $a(x,y)$ or $a(y,x)\in A$ with $y\in T)$ an adjacent set and will consist of all boundary nodes to set T.Set T is expanded by absorbing nodes in X' that satisfy $q'(x)\geq 0$ where:

$q'(x)=(U(x,T)-L(T,x))-(U(\overline{T}-x,x)-L(x,\overline{T}-x))$,where,

$U(x,T)$ represents the sum of arc upper bound or capacity values of those arcs going from x to T.

$U(\overline{T}-x,x)$ is the sum of upper bounds values of arcs going from $(\overline{T}-x)$ to x.

$L(T,x)$ is the sum of lower bound values of arcs going from T to x.

$L(x,\overline{T}-x)$ is the sum of lower bounds values of arcs going from x to $(\overline{T}-x)$.

It should be noted that $q'(x)$ is the change in $K(\overline{T},T)$ as defined in (3.1), that would occur if node x were added to T. The action taken on any node x during the backward procedure is dependent on $q'(x)$. Corresponding conditions for $q'(x)$ and appropriate actions taken are summarized as follows:

(a) $q'(x)\geq 0$, Add x to T then T is expanding and developing a new cut $(\overline{T}-x, T+x)$. Reduce previous $K(\overline{T},T)$ by $q'(x)$ which is the new minimal cut value. Redefine X', and repeat procedure using the now expanded set T.

(b) $q'(x) < 0$, another $x \in X'$ is considered. If no $x \in X'$ can be found with $q'(x) \geq 0$, then the backward procedure would terminate.

Compute the value of $K^*(s,t)$ by following conditions :

(a) If $\overline{T} = s$, then the complete procedure terminates and a minimal cut value $K^*(s,t)$ is equal to the last value of $K(\overline{T},T)$.

(b) If $\overline{T} \neq s$, then the forward procedure must be initiated which finds the value of $K^*(s,t)$

Forward procedure

The forward procedure is initiated by establishing sets W and S equal to the source node, that is $W = S = s$. $K(S,\overline{S})$ is computed by (3.2) using the source node as set S .

$$K(S,\overline{S}) = U(S,\overline{S}) - L(\overline{S},S) \qquad \text{....... (3.2)}$$

Allocate : $f(s,i) = u(s,i)$ and $f(i,s) = l(i,s)\,\forall (s,i),(i,s) \in A$

Set X is defined as an adjacent set of any set W and will consist of all boundary nodes for every set W . Set X is defined as:

$$X = (x \mid x \in \overline{W}, x \notin T, \text{and}\, a(w,x)\, \text{or}\, a(x,w) \in A\, \text{with}\, w \in W).$$

Set W is expanded by adding nodes from set X . At each iteration a node from set X is added to set W. Set W is expanded until $W = (N - T)$ at which point termination will occur. Set S will be equal or contained in W and will define the minimal cut. The most efficient forward procedure is when S=W and hence the minimal cut is defines at set W. This is similar to the backward procedure which used set T to define the minimal cut as it expands from a sink node.

Set W and S are expanded by absorbing nodes in X which is examined by $q(x)$ where:

$$q(x) = (F(W,x) - L(x,W)) - (R(x,\overline{W} - x) - L(\overline{W} - x, x))$$

$q(x)$ is the difference between impinging flow (that is flow reaching x from W) and residual capacity (that is additional capacity on arc going from x to $\overline{W} - x$) less lower bound requirements.

$F(W,x)$ is the sum of flow in all arcs from W to x.

$R(x,\overline{W}-x)$ is the sum of residual capacities of arcs out of x into $(\overline{W}-x)$.This residual is the difference between arc capacities and any allocated flow to an arc as given by $r(i,j)=u(i,j)-f(i,j)$.

$L(x,W)$ is the sum of lower bound values of arcs going from x to W.

$L(\overline{W}-x,x)$ is the sum of lower bound values of arcs going from $(\overline{W}-x)$ to x.

$q(x)$ represents a change in $K(S,\overline{S})$ as defined in (3.2), that would occur if S is expanded and a new cut located. $K(S,\overline{S})$ is again updated at each iteration by subtracting $q(x)$ from the previous value of $K(S,\overline{S})$. Expansion of W and S is dependent not only on $q(x)$ but also on relationship between S and W and two cases arises,

(a) $S=W$

(b) $S\subset W$

For the cases when $S=W$ the conditions of $q(x)$ and corresponding actions based on these conditions are summarized as follows:

If: $q(x)\geq 0$, Allocate :

$$f(x,(\overline{W}-x))=u(x,(\overline{W}-x))\,\forall(x,(\overline{W}-x))\in A$$

$$f((\overline{W}-x),x)=l((\overline{W}-x),x)\,\forall((\overline{W}-x),x)\in A$$

Add x to W and S. Reduce $K(S,\overline{S})$ by $q(x)$. Redefine x, and repeat procedure.

If: $q(x)<0$, try another x such as $q(x)\geq 0$. If no x exists, then for an x that $q(x)<0$, Allocate:

$$f(x,(\overline{W}-x))=l(x,(\overline{W}-x))\quad \forall(x,(\overline{W}-x))\in A$$

$$f((\overline{W}-x),x)=l((\overline{W}-x),x)\quad \forall((\overline{W}-x),x)\in A$$

Allocate any flow balance reaching x to arcs from x to $(\overline{W}-x)$. Add x to W but not S. Redefine X and continue the procedure noting that now $S\subset W$.

For cases when $S \subset W$, It should be noted that the set S defines the current minimal cut and set W defines nodes that have already had the $q(x)$ applied.The conditions of $q(x)$ and corresponding actions based on these conditions are summarized as follows:

(a) $q(x) < 0,$ x is added to W ,but not to S in the same manner as when $S = W$ and $q(x) < 0,$ when $q(x) < 0,$ the addition to x to W does not achieve a cut value which is lower or at least equal to the cut value already established by set S . This simply means that additional capacity exists in arcs between x and $(\overline{W} - x)$ and hence does not represent restricting arcs. Therefore, the cut value defined by S is still minimal.

(b) $q(x) \geq 0,$ a special rerouting procedure is necessary. Then start the rerouting procedure.

Rerouting procedure is necessary to determine whether excess flow reaching x can be redirected back through nodes in $(W - S)$ to $(\overline{W} - x)$ .Arcs emanating from x into $(\overline{W} - x)$ are first saturated, that is flow is allocated to each emanating arc equal to the remaining or residual capacity. At the point x , an excess flow equal to $q(x)$ exists at x which cannot be accommodated directly from x to $(\overline{W} - x)$ Hence, an attempt is made to send $q(x)$ back through nodes in $(W - S)$ to $(\overline{W} - x)$. Rerouting consists of locating a sequence of arcs in which existing flow can be altered in such a manner as to accommodate $q(x)$. It should be noted that a forward arc, that is an arc directed into or towards x , will have any existing flow reduced. And a reverse arc will have a flow increase indicating flow is being sent away from node x . If $q(x)$ can all be rerouted, then x is added to $(W - S)$ which means the minimal cut value is still located at S and will be defined by the set of nodes consisting of S plus x and all nodes which can not be reached during rerouting. Rerouting is achieved by considering set $(\overline{W} - x)$ as a single source with amount $q(x)$. A labelling type procedure is applied to send $q(x)$ through a network consisting of only $(W - S)$ nodes to a sink node which will be defined as node x. Following this labelling type procedure, unlabelled nodes would represent nodes which could not be reached and hence constitute a new minimal cut.

If rerouting is not possible, then a new minimal cut value is located which is more minimal or at least equal to the value already defined at S by adding all unlabeled nodes in $(W - S)$ and x to set S . This expanded set S will now define a new minimal cut. When rerouting is not possible, the new minimal cut is located by adding all unlabeled nodes in $(W-S)$ and x to set S. This expanded set S will now define a new minimal cut.

The following summarizes the required action when $S \subset W$:

(a)If $q(x) \geq 0$, Allocate

$$f((\overline{W}-x),x) = l((\overline{W}-x),x) \quad \forall((\overline{W}-x),x) \in A$$

$$f(x,(\overline{W}-x)) = u(x,(\overline{W}-x)) \quad \forall(x,(\overline{W}-x)) \in A$$

Attempt to reroute unallocated flow that cannot be accommodated directly from x to $(\overline{W}-x)$. That is, reroute $q(x)$ from x through nodes in $(W-S)$ to $(\overline{W}-x)$. If rerouting of $q(x)$ is possible, add x to W but not S. If rerouting is not possible, add x and all nodes which could not be reached during rerouting to S. Reduce $K(S,\overline{S})$ by $q(x)$. Redefine X and repeat procedure.

(b) $q(x) < 0$, Allocate

$$f(x,(\overline{W}-x)) = l(x,(\overline{W}-x)) \forall(x,(\overline{W}-x)) \in A$$

$$f((\overline{W}-x),x) = l((\overline{W}-x),x) \forall((\overline{W}-x),x) \in A$$

Then allocate balance of flow reaching x to arcs from x to $(\overline{W}-x)$. Add x to W but not S. Redefine X and repeat procedure. If no x exists then stop the procedure then find the value of $K^*(s,t)$

This cut search algorithm locates the minimal cut value, $K^*(s,t)$ .However, final flow allocations at algorithm termination do not repersent final flow assignments corresponding to the optimal $K^*(s,t)$ flow allocations made by the algorithm are made to establish a local flow conservation condition across a given node so that sets S and T can be developed. These flow allocations are not updated and hence do not constitute optimal flow assignment. This algorithm was developed to locate only the minimal cut and hence the optimal maximal flow value, but not the corresponding assigned flows through each arc. It should ,however be noted that for arcs that constitute the minimal cutest $(S,\overline{S})$, the allocated flows are in fact equal to assigned flow. It should also be noted that for all arcs connecting nodes within set $(W-S)$ this is also true. If final flow assignments for all arcs are necessary, then arcs connecting nodes with in S and T must have any allocated flows adjusted. This adjustment could be accomplished by sending flows allocated to boundary arcs of S and T back to the source or sink respectively.

3.3. NUMERICAL EXAMPLE

The cut search algorithm developed in the preceding section will be illustrated by applying this algorithm to the flow network given in Figure 3.1. In this Figure, arc parameters are given by (l,u). For each step, sets T , S and W will be shown along wih the computations of $q'(x)$ and $q(x)$, as well as corresponding flow allocations.

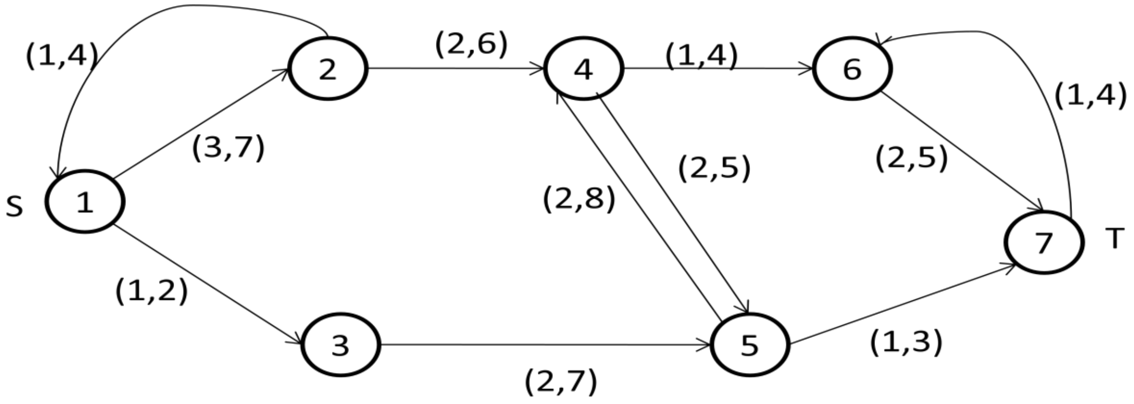


Figure 3.1

Step 1: Backward procedure initialization, $T=[7]$;

$$K(\overline{T},T)=U(\overline{T},T)-L(T,\overline{T})$$

$$=(u(6,7)+u(5,7))-l(7,6)$$

$$=(5+3)-1=7$$

Step 2: $T=[7],\ X'=[5,6],$ set $x=6;$ then

$$q'(6)=(U(6,T)-L(T,6))-(U(\overline{T}-6,6)-L(6,\overline{T}-6))$$

$$=(u(6,7)-l(7,6))-(u(4,6)-0)$$

$$= (5-1)-(4-0)=0$$

Hence add node 6 to T and $K(\overline{T},T)=7-0=7$.

Step 3: $T=[7,6],\ X'=[4,5],$ set $x=5;$ then

$$q'(5)=\ (U(5,T)-L(T,5))-(U(\overline{T}-5,5)-L(5,\overline{T}-5))$$

$$=(u(5,7)-0)-((u(3,5)+u(4,5))-l(5,4))$$

$$=(3-0)-((7+5)-2)=-7$$

Hence, try another x.

Step 4: Set $x=4;$ then

$$q'(4)=(U(4,T)-L(T,4))-(U(\overline{T}-4,4)-L(4,\overline{T}-4))$$

$$=(u(4,6)-0)-((u(2,4)+u(5,4))-l(4,5))$$

$$=(4-0)-((6+8)-2)=-8$$

Note, no other $x\in X'$ exists such that $q'(x)\geq 0$; hence terminate backward procedure.

Step 5: Forward procedure initialization, $W=S=[1]$

$$K(S,\overline{S})=U(S,\overline{S})-L(\overline{S},S)$$

$$=(u(1,2)+u(1,3))-l(2,1)$$

$$=(7+2)-1=8$$

Allocate flow: $f(1,2)=u(1,2)=7$, $f(1,3)=u(1.3)=2$, $f(2,1)=l(2,1)=1$.

Step 6 : $W=S=[1],\ \ X=[2,3],$ set $x=2;$ then

$$q(2)=(F(W,2)-L(2,W))-(R(2,\overline{W}-2)-L(\overline{W}-2,2))$$

$$=(f(1,2)-l(2,1))-(r(2,4)-0)$$

$$=(7-1)-(6-0)=0$$

Allocate flow $f(2,4)=u(2,4)=6$. Add node 2 to W and S and $K(S,\overline{S})=8-0=8$.

Step 7: $W=S=[1,2],\ X=[3,4]$, set $x=3$; then

$$q(3)=(F(W,3)-L(3,W))-(R(3,\overline{W}-3)-L(\overline{W}-3,3))$$

$$=(f(1,3)-0)-(r(3,5)-0)$$

$$=(2-0)-(7-0)=-5$$

Hence, try another x. Set $x=4$; then

$$q(4)=(F(W,4)-L(4,W))-(R(4,\overline{W}-4)-L(\overline{W}-4,4))$$

$$=(f(2,4)-0)-(r(4,6)+r(4,5))-l(5,4))$$

$$=(6-0)-((4+5)-2)=-1$$

Note: No other $x \in X$ exists such that $q(x) \geq 0$ hence add node 4 to W but not S.
Allocate flow: $f(5,4)=l(5,4)=2$, $f(4,6)=l(4,6)=1$, $f(4,5)=l(4,5)=2$, then
Allocate any flow balance: hence, $f(4,6)=3$ and $f(4,5)=5$ (note we are one unit short of capacity on arc (4,6) because $q(4)=-1$)

Step 8: $W=[1,2,4],\ S=[1,2],\ X=[3,5]$, set $x=3$; then

$$q(3)=(F(W,3)-L(3,W))-(R(3,\overline{W}-3)-L(\overline{W}-3,3))$$

$$=(f(1.3)-0)-(r(3,5))-0)$$

$$=(2=0)-(7-0)=-5$$

Allocate flow : $f(3,5)=l(3,5)=2$. Note no additional balance exists. Add node 3 to W but not S.

Step 9: $W=[1,2,4,3],\ S=[1,2],\ X=[5]$, set $x=5$; then

$$q(5)=(F(W,5)-L(5,W))-(R(5,\overline{W}-5)-L(\overline{W}-5,5))$$

$$=(f(3,5)+f(4,5)-l(5,4))-(r(5,7)-0)$$

$= ((2+5)-2)-(3-0) = 2$

Allocate flow: $= f(5,7) = r(5,7) = 3$. Attempt to reroute the $q(5) = 2$ $= f(5,7) = r(5,7) = 3$

Step 10: Rerouting: Two units, $q(5) = 2$, must be rerouted from node 5 through nodes in $(W - S)$ to $(\overline{W} - x)$ if possible. Noting Figure 3.2 can be observed that the sequence of arcs $(4,5)$ to $(4,6)$ can accommodate one unit. Hence, one unit is sent from node 5 through node 4 to node 6. Arc $(4,5)$ is a forward arc on this path, Hence, $f(4,5)$ is reduced from 5 to 4.And arc $(4,6)$ is a reverse arc on this path, hence $f(4,6)$ is increased from 3 to 4 units. Therefore $f(4,5) = 4$ and $f(4,6) = 4$. At this point $q(5) = 1$: This one unit of flow should be rerouted; However, a path cannot be located to accommodate this unit because now $r(4,6) = 0$. Therefore add node 5, add nodes 3 and 4 to S and W.

$K(S,\overline{S}) = 8 - 1 = 7$

Step 11: $W = [1,2,4,3,5]$, $S = [1,2,4,3,5]$.Note $\overline{W} = [6,7] = T$ hence terminate.

Solution to this problem is then: $K^*(s,t) = K^*(1,7) = K(S,\overline{S}) = 7$. That is, the minimal cut value is 7, hence the maximal flow from node 1 to node 7 is 7 units. The minimal cut set, as defined by $S = [1,2,3,4,5]$, is $[(4,6),(5,7)]$.This cut set represents the restricting set of network arcs. Therefore, in order to increase network flow arc $(4,6)$ and $(5,7)$ must be changed.

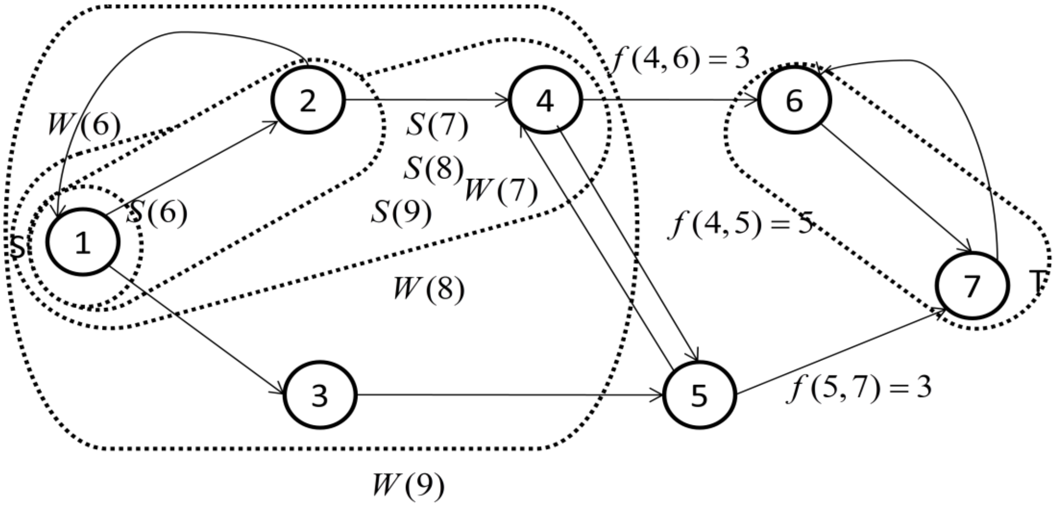


Figure 3.2 step 10- The *W*(*i*) and *S*(*i*) Indicate the Sets *W* and *S* as of step *i* of the procedure.

3.4 CONCLUSION

The cut search algorithm is applicable on networks in which arc is restricted by upper and lower flow constraints. The algorithm represented in this chapter represents an alternative approach to the maximal flow problem which has particular application when the minimal cut set arcs must be located as well as the minimal cut value. In order to locate the minimal cut set using an out-of-kilter algorithm, the algorithm first be used to establish arc flow assignments on all network arcs. Then this list must be checked to establish those arcs which have flow assignments at the stated capacity. However, using this new algorithm these arcs would be established directly by the algorithm as given by set S .

Chapter 4

ON CRITICAL CAPACITY OF ARC IN UNDIRECTED NETWORK

In this chapter an attempt has been made to find the critical capacity of arcs and maximal flow in a undirected network using the algorithm given in chapter 2.

Consider a network Phillips et. al. (1979) assumes that the original network is feasible, that is $U(W,\overline{W}) \geq L(\overline{W},W)$ for all $W \subseteq (N-T)$.If feasibility is questionable; then the necessary and sufficient conditions must be tested. So in this study feasibility conditions have been checked by the algorithm given in chapter 2.

NUMERICAL EXAMPLE:

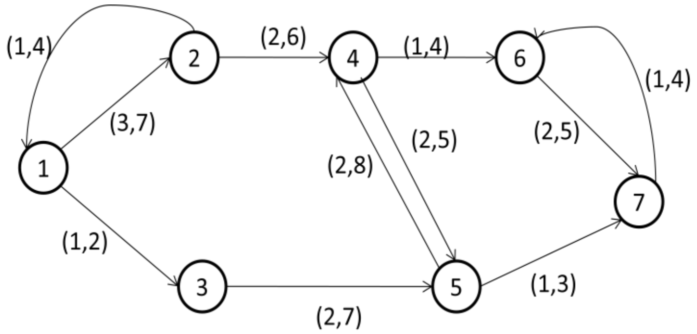


Figure 4.1

Introduce the arc (7,1) with the lower bound on the flow is zero and upper bound as infinity. Using step 2 to step 5, feasibility conditions is checked and the procedure is given from Figure 4.2 to Figure 4.5

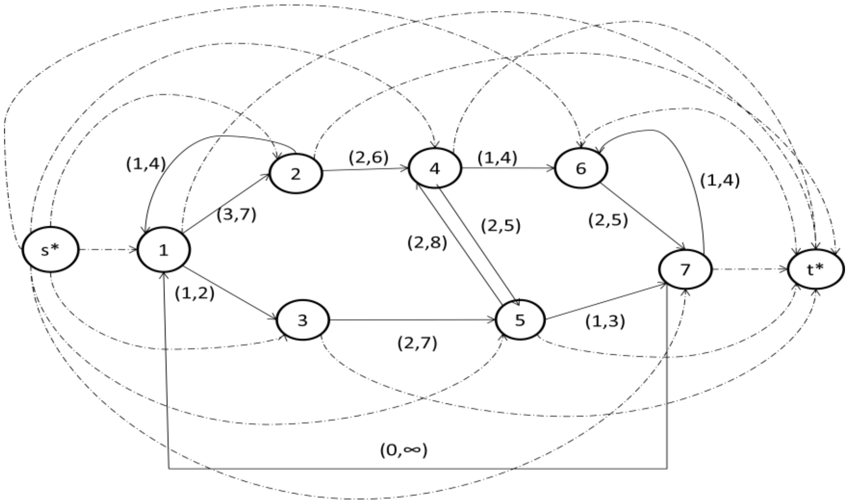


Figure 4.2

Find the value of $(l^*_{ij}, u^*_{ij}), \forall i \in N$ are given in Table 4.1 and shown in Figure 4.3

Table 4.1

(1,2)	(l^*_{12}, u^*_{12})	(0,4)
(2,1)	(l^*_{21}, u^*_{21})	(0,3)
(1,3)	(l^*_{13}, u^*_{13})	(0,1)
(2,4)	(l^*_{24}, u^*_{24})	(0,4)
(3,5)	(l^*_{35}, u^*_{35})	(0,5)
(4,5)	(l^*_{45}, u^*_{45})	(0,3)
(4,6)	(l^*_{46}, u^*_{46})	(0,3)
(5,4)	(l^*_{54}, u^*_{54})	(0,6)
(5,7)	(l^*_{57}, u^*_{57})	(0,2)
(6,7)	(l^*_{67}, u^*_{67})	(0,3)
(7,6)	(l^*_{76}, u^*_{76})	(0,3)

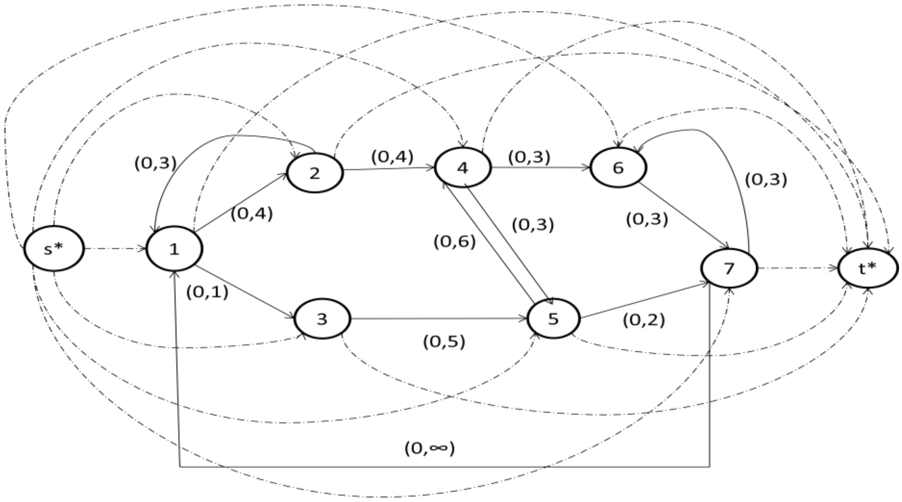


Figure 4.3

Find the value of $(l^*_{s^*i}, u^*_{s^*i}), \forall i \in N$ are given in Table 4.2 and shown in Figure 4.4

Table 4.2

$(l^*_{s^*1}, u^*_{s^*1})$	$(l^*_{s^*2}, u^*_{s^*2})$	$(l^*_{s^*3}, u^*_{s^*3})$	$(l^*_{s^*4}, u^*_{s^*4})$	$(l^*_{s^*5}, u^*_{s^*5})$	$(l^*_{s^*6}, u^*_{s^*6})$	$(l^*_{s^*7}, u^*_{s^*7})$
(0,1)	(0,3)	(0,1)	(0,4)	(0,4)	(0,2)	(0,3)

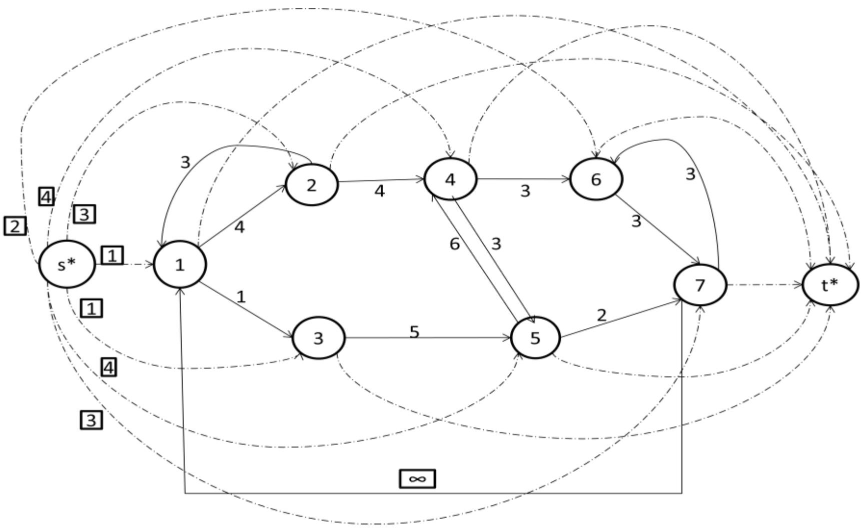


Figure 4.4

Find the value of $(l^*_{it^*}, u^*_{it^*}), \forall i \in N$ are given in Table 4.3 and shown in Figure 4.5

Table 4.3

$(l^*_{1t^*}, u^*_{1t^*})$	$(l^*_{2t^*}, u^*_{2t^*})$	$(l^*_{3t^*}, u^*_{3t^*})$	$(l^*_{4t^*}, u^*_{4t^*})$	$(l^*_{5t^*}, u^*_{5t^*})$	$(l^*_{6t^*}, u^*_{6t^*})$	$(l^*_{7t^*}, u^*_{7t^*})$
(0,4)	(0,3)	(0,2)	(0,3)	(0,3)	(0,2)	(0,1)

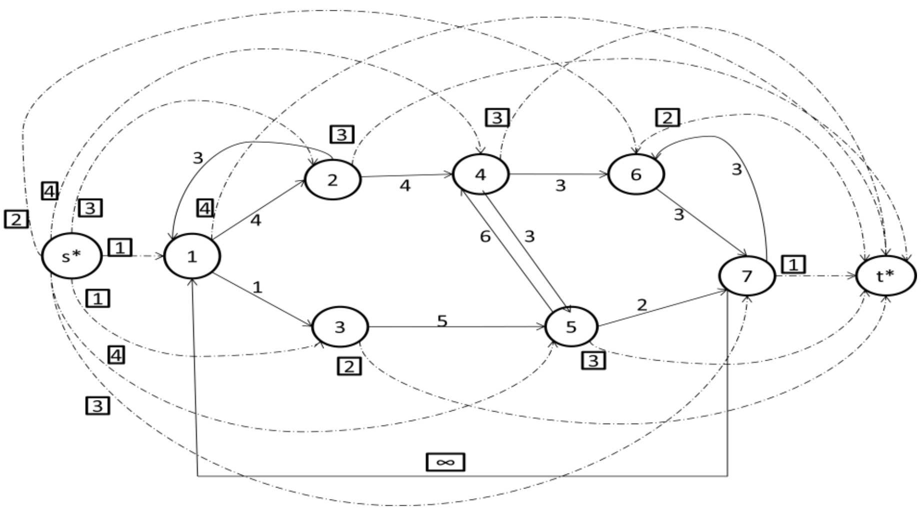


Figure 4.5

Using step 5, we find the value of V from Figure 4.5

Table 4.4

Paths	Flow
$s^* - 5 - t^*$	3
$s^* - 4 - t^*$	3
$s^* - 2 - t^*$	3
$s^* - 7 - 6 - t^*$	2
$s^* - 6 - 7 - t^*$	1
$s^* - 6 - 7 - 1 - t^*$	1
$s^* - 4 - 6 - 7 - 1 - t^*$	1
$s^* - 1 - t^*$	1
$s^* - 3 - t^*$	1
$s^* - 5 - 7 - 1 - t^*$	1
$s^* - 7 - 1 - 3 - t^*$	1
	$V = 18$

Table 4.4 shows that $V = \sum_{(i,j)\in A} l_{ij}$, so the network is feasible.

Using step 6, find the critical capacity and modified lower bound of each arc of given network G^* given in Figure 4.5. and the procedure has been shown from Figure 4.6(a) to Figure 4.7(b) and in Table 4.5(a) to 4.5(b) for each arc.

Arc (1,2)

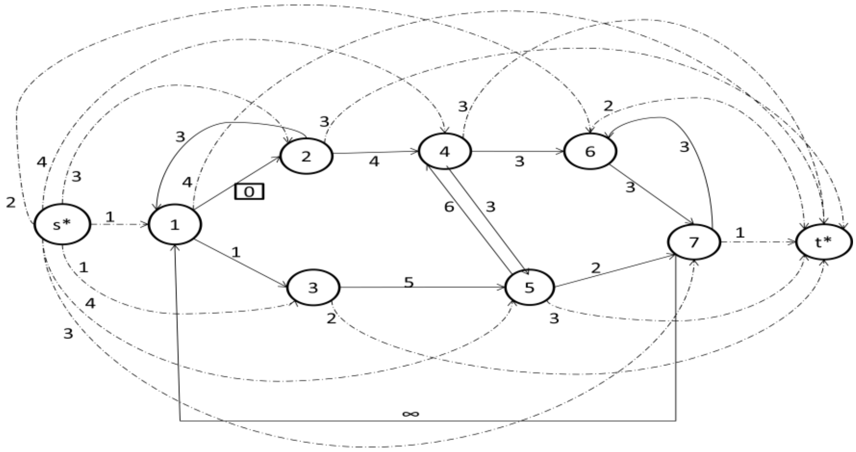


Figure 4.6(a)

When the capacity of (1,2) arc is zero is shown in Figure 4.6(a).Then maximum flow from source to sink is determined by following paths which are given in Table 4.5(a).

Table 4.5(a)

Paths	Flow
s^*-5-t^*	3
s^*-4-t^*	3
s^*-2-t^*	3
$s^*-7-6-t^*$	2
$s^*-6-7-t^*$	1
$s^*-6-7-1-t^*$	1
$s^*-4-6-7-1-t^*$	1
s^*-1-t^*	1
s^*-3-t^*	1
$s^*-5-7-1-t^*$	1
$s^*-7-1-3-t^*$	1
	$v(0)=18$

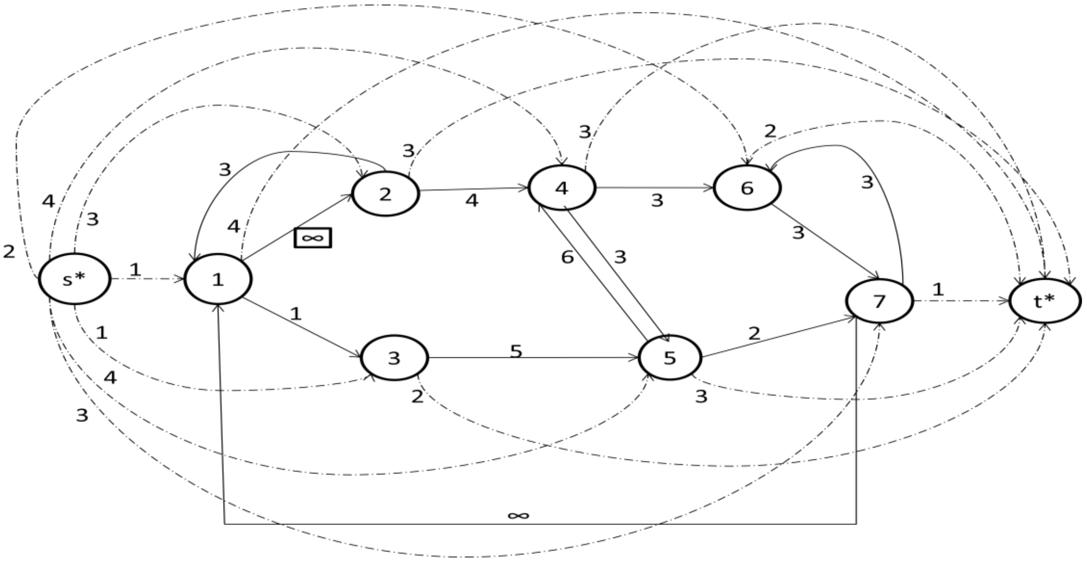


Figure 4.6(b)

When the capacity of (1,2) arc is infinity is shown in Figure 4.6(b).Then maximum flow from source to sink is determined by following paths which are given in Table 4.5(b).

Table 4.5(b)

Paths	Flow
$s^* - 5 - t^*$	3
$s^* - 4 - t^*$	3
$s^* - 2 - t^*$	3
$s^* - 7 - 6 - t^*$	2
$s^* - 6 - 7 - t^*$	1
$s^* - 6 - 7 - 1 - t^*$	1
$s^* - 4 - 6 - 7 - 1 - t^*$	1
$s^* - 1 - t^*$	1
$s^* - 3 - t^*$	1
$s^* - 5 - 7 - 1 - t^*$	1
$s^* - 7 - 1 - 3 - t^*$	1
	$v(\infty) = 18$

Then $k^*{}_{12}$ (Critical capacity of arc $(1,2)$ with respect to G^*) = $v(\infty) - v(0) = 18 - 18 = 0$

Hence $\hat{l}_{12}$ (Modified lower bound on the flow on (1,2) in G)$= l_{12} + k^*{}_{12} = l_{12} + 0 = l_{12}$

Arc (2,1)

Then $k^*{}_{21}$ (Critical capacity of arc $(2,1)$ with respect to G^*) = $v(\infty) - v(0) = 18 - 18 = 0$

Hence $\hat{l}_{21}$ (Modified lower bound on the flow on (2,1) in G) $= l_{21} + k^*{}_{21} = l_{21} + 0 = l_{21}$.

Arc (1,3)

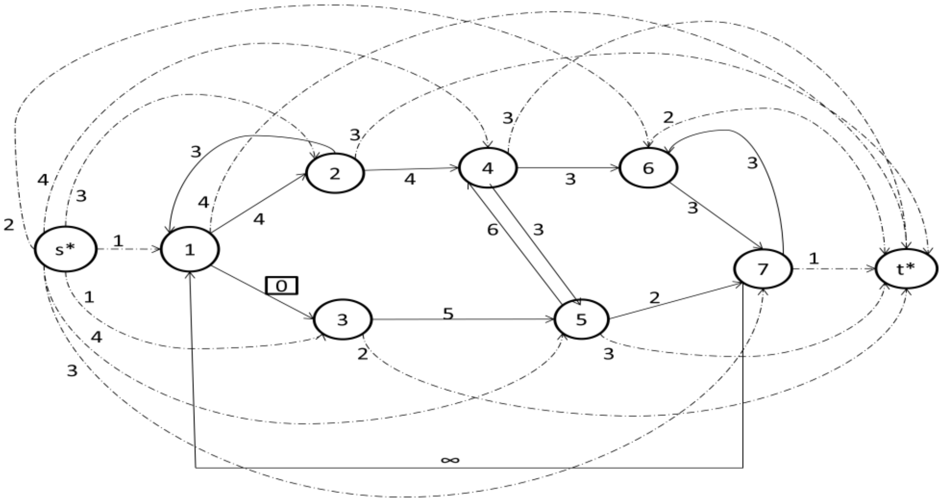


Figure 4.7(a)

When the capacity of (1,3) arc is zero is shown in Figure 4.7(a). Then maximum flow from source to sink is determined by following paths which are given in Table 4.6(a).

Table 4.6(a)

Paths	Flow
$s^* - 5 - t^*$	3
$s^* - 4 - t^*$	3
$s^* - 2 - t^*$	3
$s^* - 7 - 6 - t^*$	2
$s^* - 6 - 7 - t^*$	1
$s^* - 6 - 7 - 1 - t^*$	1
$s^* - 4 - 6 - 7 - 1 - t^*$	1
$s^* - 1 - t^*$	1
$s^* - 3 - t^*$	1
$s^* - 5 - 7 - 1 - t^*$	1
	$v(0) = 17$

`

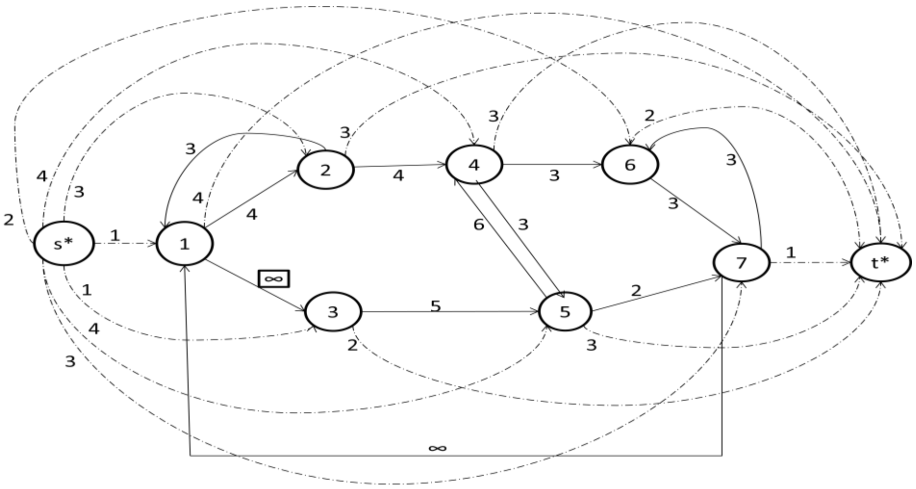


Figure 4.7(b)

When the capacity of (1,3) arc is infinity is shown in Figure 4.7(b).Then maximum flow from source to sink is determined by following paths which are given in Table 4.5(b).

Table 4.6(b)

Paths	Flow
$s^* - 5 - t^*$	3
$s^* - 4 - t^*$	3
$s^* - 2 - t^*$	3
$s^* - 7 - 6 - t^*$	2
$s^* - 6 - 7 - t^*$	1
$s^* - 6 - 7 - 1 - t^*$	1
$s^* - 4 - 6 - 7 - 1 - t^*$	1
$s^* - 1 - t^*$	1
$s^* - 3 - t^*$	1
$s^* - 5 - 7 - 1 - t^*$	1
$s^* - 7 - 1 - 3 - t^*$	1
	$v(\infty) = 18$

Then ${k^*}_{13}$ (critical capacity of arc $(1,3)$ with respect to G^*) = $v(\infty)-v(0)=18-17=1$

Hence $\hat{l}_{13}$ (modified lower bound on the flow on (1,3) in G) $=l_{13}+{k^*}_{13}=l_{13}+0=l_{13}$.

Arc (2,4)

Then ${k^*}_{24}$ (critical capacity of arc $(2,4)$ with respect to G^*) = $v(\infty)-v(0)=18-18=0$

Hence $\hat{l}_{24}$ (modified lower bound on the flow on (2,4) in G) $=l_{24}+{k^*}_{24}=l_{24}+0=l_{24}$.

Arc (3,5)

Then ${k^*}_{35}$ (critical capacity of arc $(3,5)$ with respect to G^*) = $v(\infty)-v(0)=18-18=0$

Hence $\hat{l}_{35}$ (modified lower bound on the flow on (3,5) in G) $=l_{35}+{k^*}_{35}=l_{35}+0=l_{35}$.

Arc (4,5)

Then ${k^*}_{45}$ (critical capacity of arc $(4,5)$ with respect to G^*) = $v(\infty)-v(0)=18-18=0$

Hence $\hat{l}_{45}$ (modified lower bound on the flow on (4,5) in G)$=l_{45}+{k^*}_{45}=l_{45}+0=l_{45}$

Arc (4,6)

Then ${k^*}_{46}$ (critical capacity of arc (4,6) with respect to G^*) = $v(\infty)-v(0)=18-18=0$

Hence $\hat{l}_{46}$ (modified lower bound on the flow on (4,6) in G) $=l_{46}+{k^*}_{46}=l_{46}+0=l_{46}$

Arc (5,4)

Then ${k^*}_{54}$ (critical capacity of arc (5,4) with respect to G^*) = $v(\infty)-v(0)=18-18=0$

Hence $\hat{l}_{54}$ (modified lower bound on the flow on (5,4) in G)$=l_{54}+{k^*}_{54}=l_{54}+0=l_{54}$

Arc (5,7)

Then ${k^*}_{57}$ (critical capacity of arc (5,7) with respect to G^*) = $v(\infty)-v(0)=18-18=0$

Hence $\hat{l}_{57}$ (modified lower bound on the flow on (5,7) in G) $=l_{57}+{k^*}_{57}=l_{57}+0=l_{57}$

Arc (6,7)

Then ${k^*}_{67}$ (critical capacity of arc (6,7) with respect to G^*) = $v(\infty)-v(0)=18-18=0$

Hence $\hat{l}_{67}$ (modified lower bound on the flow on (6,7) in G)$= l_{67} + k^{*}_{67} = l_{67} + 0 = l_{67}$

Arc (7,6)

Then k^{*}_{76} (critical capacity of arc (7,6) with respect to G^{*}) = $v(\infty) - v(0) = 18 - 18 = 0$

Hence $\hat{l}_{76}$ (modified lower bound on the flow on (7,6) in G)$= l_{76} + k^{*}_{76} = l_{76} + 0 = l_{76}$

Using Step 8 the flow on augmented network is shown in the Figure 4.8 in small square on each arc.

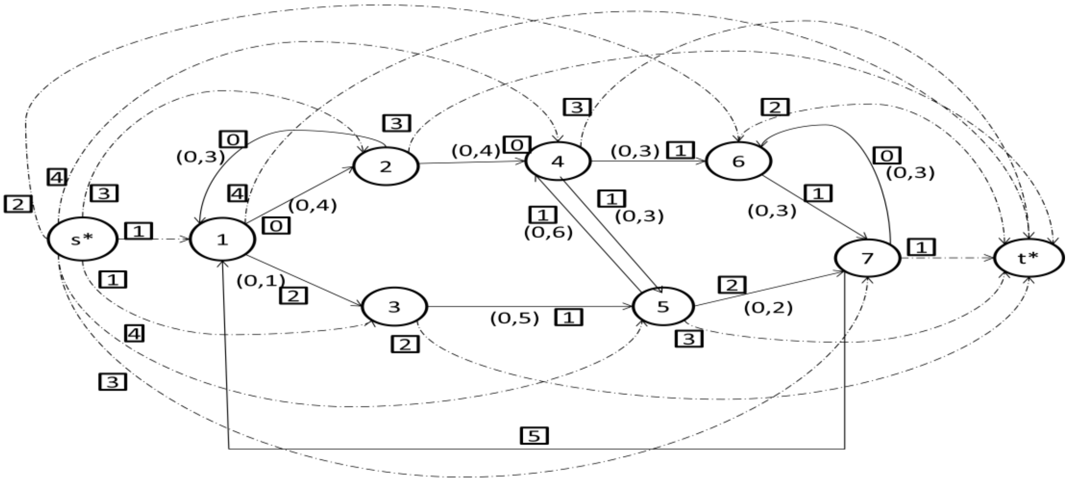


Figure 4.8

The amount of flow on each arc in a flow vector is marked inside a small square by the side of the arc, verify that f is feasible [e.g.,total amount reaching 7 is 6 units,1 units along (6,7), 2 units along (5,7) and 3 units along (s^{*},7).Total amount leaving 7 is 6 units along (7,1) and 1 unit along $(7,t^{*})$.so conservation holds at 7,etc.].This flow vector is 18.

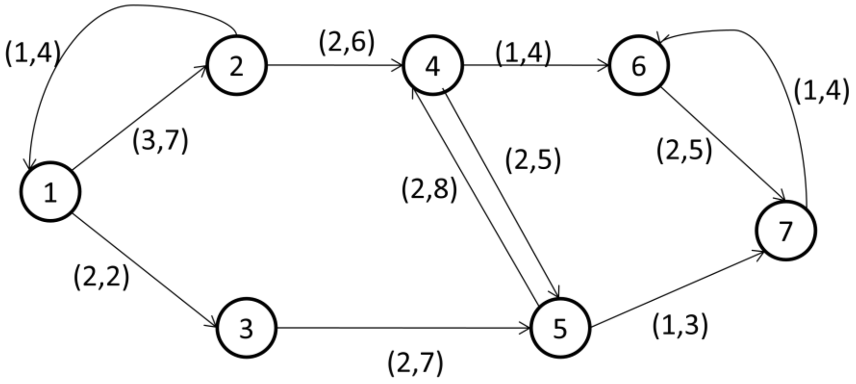


Figure 4.9

Using Step 9, the modified network G is given in Figure 4.9, the critical capacity of each arc of modified network using step 10 has been shown from Figure 4.10(a) to Figure 4.10(b) and Table 4.7 to Table 2.23 and current capacity of each arc using step 11.To check which arc is forward arc or not the step 12 is used.

Arc (1,2)

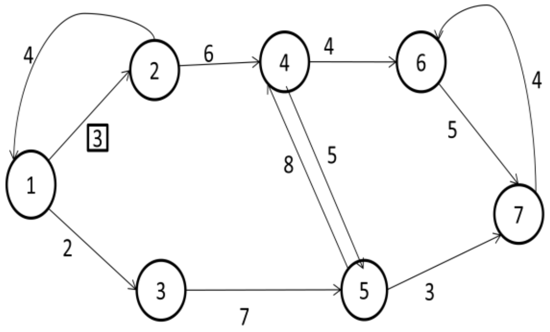


Figure 4.10(a)

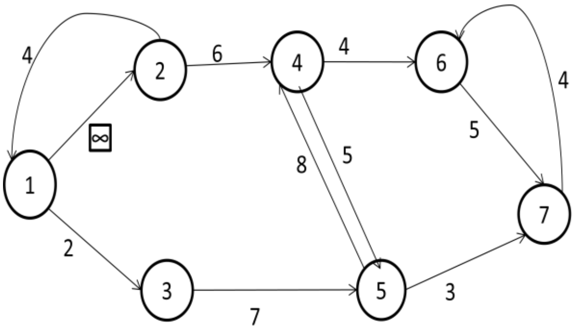


Figure 4.10(b)

When the capacity of (1,2) arc is $\hat{l}_{12}$ then maximum flow from source to sink and When the capacity of (1,2) arc is infinity then maximum flow from source to sink is determined by following paths which are given in Table 4.7.

Table 4.7

Paths	Flow	Paths	Flow
1-2-4-5-4-6-7	3	1-2-4-5-4-6-7	4
1-3-5-7	2	1-2-4-5-7	1
		1-3-5-7	2
	$v(\hat{l}_{12}) = 5$		$v(\infty) = 7$

$\hat{k}_{12}$ (Critical capacity of arc $(1,2)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{12})$

$\hat{k}_{12} = 7 - 5 = 2$

k_{12} (Current capacity of arc $(1,2)$ with respect to modified network G) $= 7 - 3 = 4$

Then $\hat{k}_{12} < k_{12}$

Therefore , arc $(1,2)$ is a not a forward arc.

Arc (2,1)

$\hat{k}_{21}$ (Critical capacity of arc $(2,1)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{21})$

$\hat{k}_{21} = 8 - 8 = 0$

k_{21} (Current capacity of arc $(2,1)$ with respect to modified network G) $= 4 - 1 = 3$

Then $\hat{k}_{21} < k_{21}$

Therefore , arc $(2,1)$ is a not a forward arc.

Arc (1,3)

$\hat{k}_{13}$ (Critical capacity of arc $(1,3)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{13})$

$\hat{k}_{13} = 7 - 7 = 0$

k_{13} (Current capacity of arc $(1,3)$ with respect to modified network G) $= 2 - 2 = 0$

Then $\hat{k}_{13} = k_{13}$

Therefore, arc $(1,3)$ not a forward arc.

Arc (2,4)

$\hat{k}_{24}$ (Critical capacity of arc $(2,4)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{24})$

$\hat{k}_{24} = 7 - 4 = 3$

k_{24} (Current capacity of arc $(2,4)$ with respect to modified network G) $= 6 - 2 = 4$

Then $\hat{k}_{24} < k_{24}$

Therefore, arc $(2,4)$ is not a forward arc.

Arc (3,5)

$\hat{k}_{35}$ (Critical capacity of arc $(3,5)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{35})$

$\hat{k}_{35} = 7 - 6 = 1$

k_{35} (Current capacity of arc $(3,5)$ with respect to modified network G) $= 7 - 2 = 5$

Then $\hat{k}_{35} < k_{35}$

Therefore, arc $(3,5)$ is not a forward arc.

Arc (4,5)

$\hat{k}_{45}$ (Critical capacity of arc $(4,5)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{45})$

$\hat{k}_{45} = 7 - 7 = 0$

k_{45} (Current capacity of arc $(4,5)$ with respect to modified network G) $= 5 - 2 = 3$

Then $\hat{k}_{45} < k_{45}$

Therefore, arc $(4,5)$ is not a forward arc.

Arc (4,6)

$\hat{k}_{46}$ (Critical capacity of arc $(4,6)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{46})$

$\hat{k}_{46} = 7 - 4 = 3$

k_{46} (Current capacity of arc $(4,6)$ with respect to modified network G) $= 4 - 1 = 3$

Then $\hat{k}_{46} = k_{46}$

Therefore, arc $(4,6)$ is a forward arc.

Arc (5,4)

$\hat{k}_{54}$ (Critical capacity of arc $(5,4)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{54})$

$\hat{k}_{54} = 7 - 7 = 0$

k_{54} (Current capacity of arc $(5,4)$ with respect to modified network G) $= 8 - 2 = 6$

Then $\hat{k}_{54} < k_{54}$

Therefore, arc $(5,4)$ is not a forward arc.

Arc (5,7)

$\hat{k}_{57}$ (Critical capacity of arc $(5,7)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{57})$

$\hat{k}_{57} = 8 - 5 = 3$

k_{57} (Current capacity of arc $(5,7)$ with respect to modified network G) $= 3 - 1 = 2$

Then $\hat{k}_{57} > k_{57}$

Therefore, arc $(5,7)$ is a forward arc.

Arc (6,7)

$\hat{k}_{67}$ (Critical capacity of arc $(6,7)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{67})$

$\hat{k}_{67} = 7 - 5 = 2$

k_{67} (Current capacity of arc $(6,7)$ with respect to modified network G) $= 5 - 2 = 3$

Then $\hat{k}_{67} < k_{67}$

Therefore, arc $(6,7)$ is not a forward arc.

Arc (7,6)

$\hat{k}_{76}$ (Critical capacity of arc $(7,6)$ with respect to modified network G) $= v(\infty) - v(\hat{l}_{76})$

$\hat{k}_{76} = 7 - 7 = 0$

k_{76} (Current capacity of arc $(7,6)$ with respect to modified network G) $= 4 - 1 = 3$

Then $\hat{k}_{76} < k_{76}$

Therefore, arc $(7,6)$ is not a forward arc.

Uing step 13, we find the max flow shown in Figure 4.11 and value is 7.

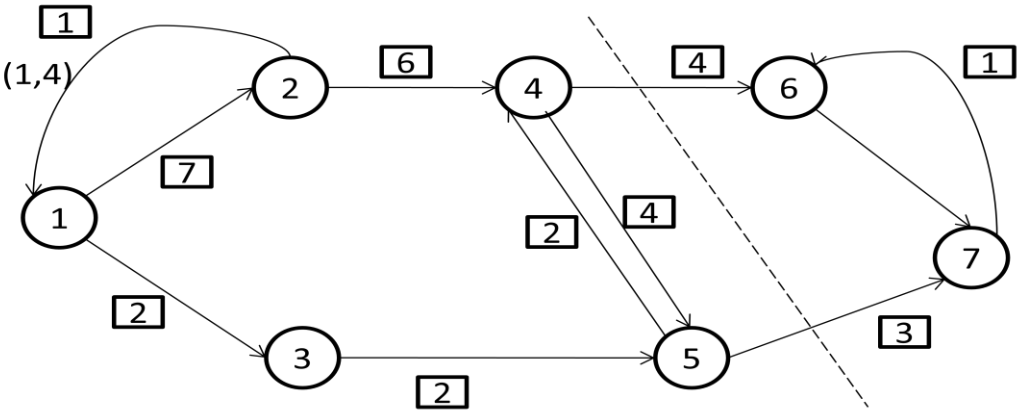


Figure 4.11

REFERENCES

[1] MURTY KATTA G., “Linear and combinatorial programming”, Department of industrial and operation engineering university of Michigan.

[2] Ahuja Ravindra K. , Magnanti Thomas L., Orlin James B., Sloan W.P., “Network Flows”, No. 2059-88 August 1988 Revised: December, 1988

[3] Sonia, Puri M.C., “On critical capacity of arcs in a directed network”, Applied mathematics and computation, Vol. 181, No. 2, pp. 1568-1576, 2006.

[4] Bazaraa M.S., Jarvis J., Sherali H.D., “Linear programming and network flows, second edition”, Wiley India.

[5] “Network Flow III – Applications By Sariel Har-Peled, Chapter 14, 2007, (valis.cs.uiuc.edu/~sariel/teach/notes/algos/lec/14_flow_III.pdf)

[6] Phillips S. and Dessouky M. I., “The cut search algorithm with arc capacities and lower bounds”, Management Science Vol. 25, no. 4, 1979.

[7] Krause Stefan, “Increasing Distances in Graphs”, 2006.

[8] Swanson H. S. and Woolsey R. E. D., “An "Out-of-Kilter" network tutorial”, Newsletter Issue 13, January 1973 **.**

[9] Ahuja R. K., Magnanti T. L, and Orlin J. B., “Network Flows Theory, Algo-rithms, and Applications”, Prentice Hall, Englewood Clis, NJ, 1993.

[10] A Primer, E. P. Durbin and D. M. Kroenke, “The Out-of-Kilter Algorithm” December 1967.

[11] Joseph S., “The Max –flow Min cut Theorem”, December 6, 2007.

[12] Cheriyan J., Mehlhorn K., “An Analysis of the Highest-Level Selection Rule in the Preflow-Push Max-Flow Algorithm” Vol. 69, pp-239-242, 1998.

[13] Savari S. A., Kramer G., “Edge-Cut Bounds on Network Coding Rates” Network and system management, 2006.

[14] Pham T. L., “A Distributed Algorithm for the Maximum Flow Problem”, IEEE computer society Washington, 2005.

[15] Even S., Endre Tarjan R., “Network flow and testing graph connectivity”, Vol. 4, 1975.

[16] Daniel Wayne K., “Generalized maximum flow algorithm”, Ph.D Thesis, Cornell University, 1999.

[17] Ghiyasvand M., “A new approach for computing a new approach for computing a most positive cut using the minimum flow algorithms”, Applied mathematics and computation, Vol. 176(1), pp. 27-39, 2006.

[18] Bertsekas D., “Network optimization: Continuous and discrete models”, Athena Scientific, Belmont, 1998.

[19] Cook W. J., Cunningham W. H., Puleyblank W. R. and Schrijver A., “Combinatorial Optimization”, Wiley, New York, 1998.

[20] Dantzig G. B., “Linear Programming and Extensions”, Princeton University Press, Princeton, 1962.

[21] Ford L. R. and Fulkerson D. R., “Flows in Networks”, Princeton university Press, Princeton, 1962.

[22] Goldfarb D. and Lin Y., “Combinatorial interior point methods for generalized network flow problems”, Mathematical Programming, Vol. 93, pp. 227-246, 2002.

[23] Jewell W. S., “Optimal flow through networks with gains”, Operations Research, Vol. 10, pp. 476-499, 1962.

[24] Kapoor S. and Vaidya P. M., “Speeding up Karmarkar’s algorithm for multicommodity flows”, Mathematical Programming, Vol. 73, pp. 111-127, 1967.

[25] Murray S. M., “An interior point approach to the generalized flow problem with costs and related problems”, Ph.D Thesis, Stanford University, 1993.

[26] Schrijver A., “Combinatorial optimization: Polyhedra and Efficiency”, Springer, Berlin, 2003.

[27] Shigeno M., “Maximum network flow with concave gains”, Discussion paper, Vol. 1027, University of Tsukuba, 2003.

[28] Ford L. R. And Fulkerson D. R., “Maximal flow through a network”, Canadian Journal of Mathematics, Vol. 8, pp. 399-404, 1956.