The Government of the Union of Myanmar
Ministry of Education

Department of Higher Education (Lower Myanmar)
and
Department of Higher Education (Upper Myanmar)

Universities Research Journal

Vol. 1, No. 2 December, 2008.

Universities Research Journal 2008
Vol. 1, No. 2

Editorial Board

Contents

Page

Arc-Disjoint Path Pair (APP) Problem

Nang Kham Maing

Abstract

Let G (V,A) be a directed network containing *n* nodes $v \in V$ and *m* arcs $(i, j) \in A$,each with a non-negative length. In this paper arc-disjoint path pair problem is considered, which is to find a pair of arc-disjoint paths from a source to a destination in a directed network. We described Bhandari's algorithm to solve arc-disjoint path pair problem and also proved the correctness of algorithm.

Key words: Directed network, Shortest paths, Arc-disjoint path.

Introduction

Disjoint paths are used in communication networks for reliability of transmission between a given source and destination. Paths between a given pair of source and destination nodes in a network are called arc-disjoint if they have no common arcs. In this paper, we consider the problem of finding a pair of arc-disjoint paths between a pair of nodes in a directed network.

Firstly, we introduced arc-disjoint path pair problem and some basic definitions and notations are presented. Secondly, Bhandari's algorithm for solving arc-disjoint path pair problem is described. Finally, the correctness of Bhandari's algorithm is proved and we also prove that the solution produced with Bhandari's algorithm is optimal.

Statement of Arc-Disjoint Path Pair Problem

Given a network G (V, A) for a source-destination pair (s, t), find a set of two paths P_1 and P_2, such that $P_1 \cap P_2 = \varnothing$ and the total length $l(P_1)+l(P_2)$ is minimized.

Tutor, Department of Mathematics, Lashio University

Some Basic Definitions and Notations on Arc-Disjoint Path Pair Problem

An **arc** from node i to node j is represented as (i, j), i, j $\in$ V. Each arc (i, j) has a numerical length (weight) w_{ij}. We assume that only non-negative arc lengths are assigned to each arc. However, in the process of computing disjoint paths, negative arc lengths may be assigned to arcs.

Let G (V, A) be a network, a **path P**, between a source s and destination t is considered to be a set of arcs that compose this path. With a slight abuse of notation, we choose P to denote the path as well as its arc set. The length of a path l(P) is computed as $l(P) = \sum_{(i,j)\in P} w_{ij}$.

We define the **total length** of two paths as $l(P_1) + l(P_2)$.

Let P be a path from s to t, q is said to be a **sub-path** of P if it coincides with P from s until y.

The **shortest path** between two nodes i and j is the path joining i to j, that has minimum length.

The **intersection** of paths P_1 and P_2 is the set of arcs which contains both P_1 and P_2,

i.e., $P_1 \cap P_2 = \{(i,j) \in A \mid (i,j) \in P_1 \text{ and } (i,j) \in P_2\}$.

The **union** of paths P_1 and P_2 is the set of arcs which contains either P_1 or P_2, i.e., $P_1 \cup P_2 = \{(i,j) \in A \mid (i,j) \in P_1 \text{ or } (i,j) \in P_2\}$.

If path P_1 is **arc-disjoint** with P_2, there is no common arc element in the arc set representing each path and $P_1 \cap P_2 = \varnothing$, else $P_1 \cap P_2 \neq \varnothing$.

If path P_1 is arc-disjoint with P_2, i.e., $P_1 \cap P_2 = \varnothing$, we have $l(P_1 \cup P_2) = l(P_1) + l(P_2)$.

The path which is obtained by reversing the direction and the sign of the arc lengths of each arc on the path P_1 between s and t is called the path **-P₁** directed from t to s.

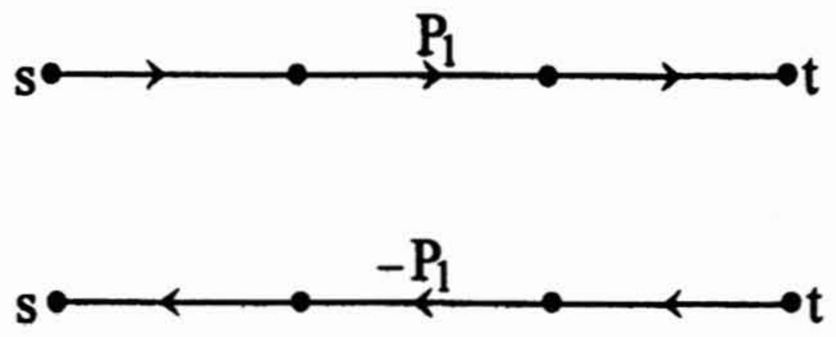

Fig. 1. Paths P_1 and $-P_1$

$\mathbf{P_1 \tilde{\cap} P_2}$ is a set, which consists of the P_1 arcs whose reversed arcs appear on P_2 and vice versa,

i.e., $P_1 \tilde{\cap} P_2 = \{(i, j) \text{ and } (j, i) | (i, j) \in P_1 \text{ and } (j, i) \in P_2\}$.

A **modified network G (V, A′)** is a network which is created by reversing the direction and the sign of the length of each arc on the shortest path P_1 from node s to node t in a directed network G (V, A).

In all the figures, **bold lines** represent arcs on the shortest path(s) in a network or its corresponding modified network, **dashed lines** represent reversed arcs which do not exist in the original network and **bold dashed lines** represent such reversed arcs that appear on the shortest path.

Bhandari's Algorithm

The steps of the Bhandari's algorithm for solving arc-disjoint path pair problem are as follows:

Given a directed network G (V, A), for a source-destination pair (s, t),

Step 1. Find the shortest path P_1 from node s to node t;

Step 2. Replace P_1 with $-P_1$, a modified network G (V, A′) is created;

Step 3. Find a shortest path P_2 from node s to node t in the modified network G (V, A′), if P_2 does not exist, then stop;

Step 4. Take the union of P_1 and P_2, remove from the union the arc set which consists of the P_1 arcs whose reversed arcs appear in P_2 and vice versa, then group the remaining arcs into two paths P_1' and P_2', i.e., $P_1' \cup P_2' = (P_1 \cup P_2) \setminus (P_1 \tilde{\cap} P_2)$

We will explain the steps of Bhandari's algorithm with examples in Fig. 2. and Fig. 3.

Example (1)

We are required to find a set of two shortest disjoint paths between s and t.

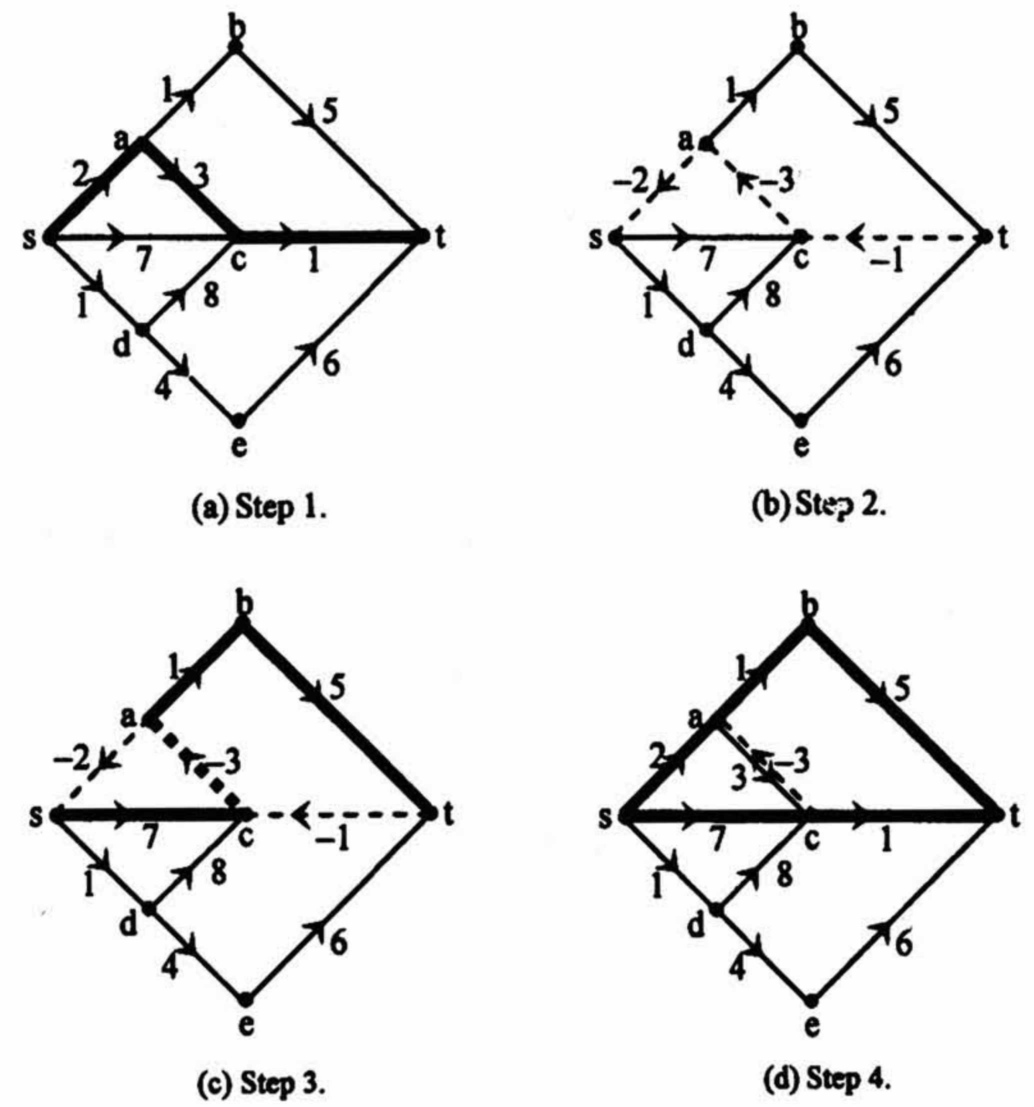

Fig. 2. Example of the operation of Bhandari's algorithm

In Step 1, the shortest path from s to t is found as $P_1 = \text{sact}$, with minimum length 6. In Step 2, a modified network G (V, A′) is created by replacing P_1 with $-P_1$. In Step 3, the shortest path in the modified network $P_2 = \text{scabt}$ has length 10. In Step 4, $P_1 \tilde{\cap} P_2 = \{(a, c), (c, a)\}$ is removed from the union $P_1 \cup P_2$. The solution set of disjoint paths $\{P_1', P_2'\} = \{\text{sabt, sct}\}$ is obtained. The total length of this path set equals 8 + 8 = 16, which is exactly the minimal total length of two arc-disjoint paths in this network.

Example (2)

Bhandari's algorithm is exemplified with the network in Fig. 3.

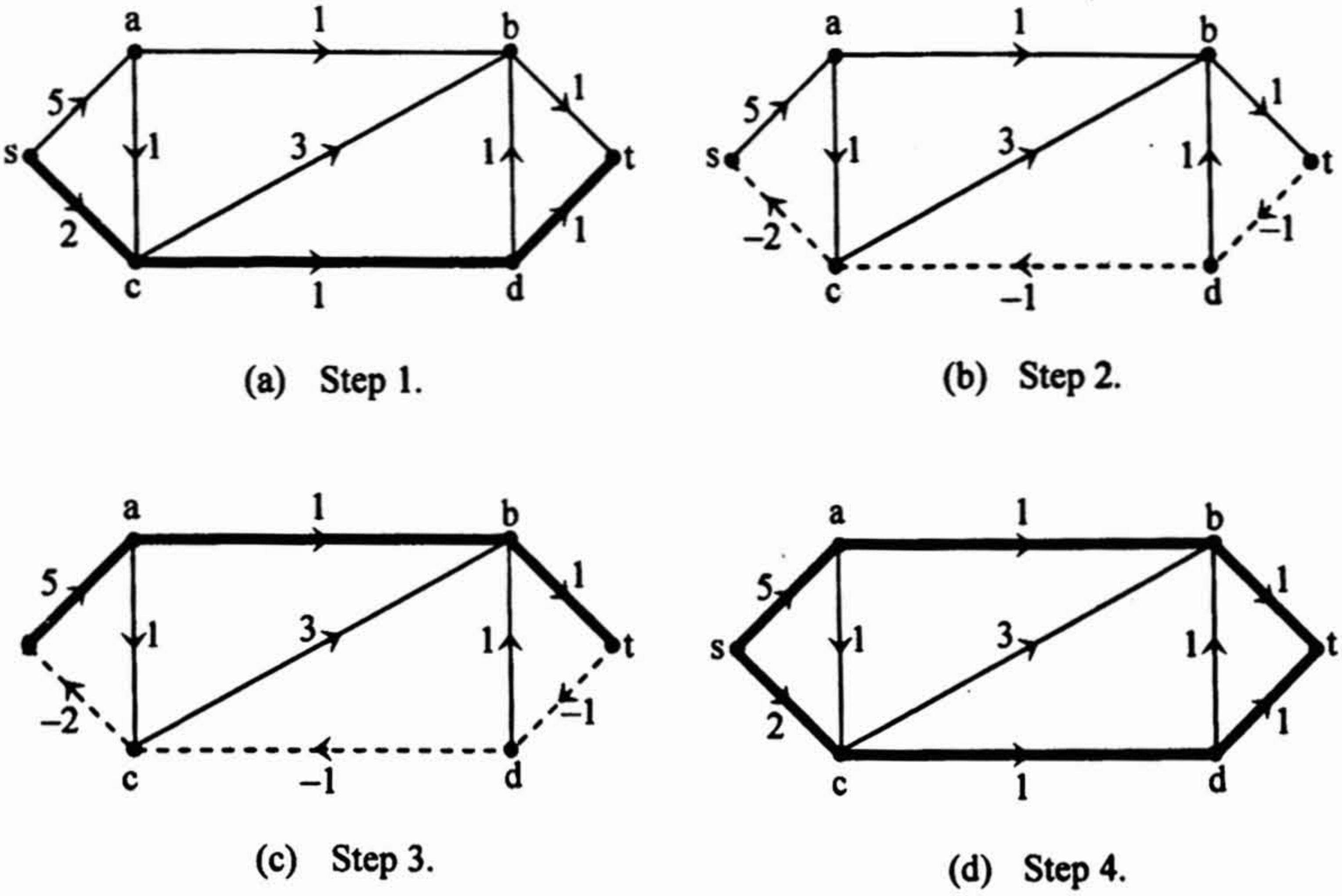

Fig. 3. Example of the operation of Bhandari's algorithm

In Step 1, the shortest path from s to t is found as $P_1 = \text{scdt}$, with minimum length 4. In Step 2, a modified network G (V, A′) is created by replacing P_1 with $-P_1$. In Step 3, the shortest path in the modified network $P_2 = \text{sabt}$ has length 7. In Step 4, $P_1 \tilde{\cap} P_2 = \varnothing$. The solution set of disjoint paths $\{P_1', P_2'\} = \{\text{scdt, sabt}\}$ is obtained. The total length of this path set equals 4 + 7 = 11 which is exactly the minimal total length of two arc-disjoint paths in this network.

Correctness of Bhandari's Algorithm

We will first show that the optimal solution set of Bhandari's algorithm is based on the shortest path.

Given a network G (V, A) and a pair of source-destination nodes (s, t), the relation between a set of two arc-disjoint paths $\{P_1', P_2'\}$ and the shortest path P_1 belongs to one of the following types.

(1) P_1 itself is P_1' or P_2', i.e., $P_1 = P_1'$ or $P_1 = P_2'$;

(2) P_1 overlaps with both paths P_1' and P_2', i.e., $P_1 \cap P_1' \neq \varnothing$, $P_1 \neq P_1'$ and $P_1 \cap P_2' \neq \varnothing$, $P_1 \neq P_2'$;

(3) P_1 only overlaps with one path in the set $\{P_1', P_2'\}$ but not with the other one, i.e., $P_1 \cap P_1' \neq \varnothing$, $P_1 \neq P_1'$ and $P_1 \cap P_2' = \varnothing$ (or) $P_1 \cap P_2' \neq \varnothing, P_1 \neq P_2'$ and $P_1 \cap P_1' = \varnothing$;

(4) P_1 is arc-disjoint with both paths in $\{P_1', P_2'\}$, i.e., $P_1 \cap (P_1' \cup P_2') = \varnothing$.

Lemma (1)

Given a directed network G (V, A) and a source-destination pair (s, t), if the optimal set $\{P_1', P_2'\}$ of APP exists, $P_1' \cup P_2'$ must contain either the first shortest path P_1 itself or some P_1 arcs on each of its two paths.

Proof.

If $P_1' \cup P_2'$ is of type (4), then each path in $\{P_1', P_2'\}$ is arc-disjoint with P_1. As P_1 is the shortest path, both $\{P_1, P_1'\}$ and $\{P_1, P_2'\}$ have a total length shorter than $\{P_1', P_2'\}$. Hence the optimal set $\{P_1', P_2'\}$ cannot be of type (4) and $P_1' \cup P_2'$ must contain some or all P_1 arcs to be the optimal set.

If $P_1' \cup P_2'$ is of type (3), only one path in $P_1' \cup P_2'$ contains some P_1 arcs, without loss of generality, suppose P_1' contains some P_1 arcs and the other path P_2' is arc-disjoint with P_1, then $\{P_1, P_2'\}$ is a set which is shorter than $\{P_1', P_2'\}$. Hence the optimal set $\{P_1', P_2'\}$ cannot be of type (3).

Therefore, if the optimal set $\{P_1', P_2'\}$ exists, $P_1' \cup P_2'$ must be either of type (1) or (2). ∎

Definition (1)

The logical difference set $P_2 - P_1$ also can be computed as

$$P_2 - P_1 = \{(i, j) \mid (i, j) \in P_2 \setminus (P_2 \cap P_1)\} \cup \{(j, i) \mid (i, j) \in P_1 \setminus (P_2 \cap P_1)\}$$

which means that if an arc (i, j) of P_2 does not appear on P_1, then this arc belongs to the difference set $P_2 - P_1$ and if an arc (i, j) of P_1 does not appear on P_2, then its direction reversed arc (j, i) belongs to the difference set $P_2 - P_1$ with an arc length $w_{ij} = -w_{ji}$.

Remark

In set theory, the difference operation is defined as $P_2 - P_1 = P_2 \setminus (P_1 \cap P_2)$ and the symmetric difference operation is defined as $P_2 - P_1 = (P_2 \cup P_1) \setminus (P_1 \cap P_2)$. The concept of logical difference set in this paper resembles the symmetric difference set but it is not the same.

Lemma (2)

Let us denote $O_l = (P_1' \cup P_2') \tilde{\cap} (-P_1)$, which means that the set O_l consists of each P_1 arc in the union of $P_2 \cup P_1$ and its corresponding $-P_1$ arc. Then $O_l = P_1' \cup P_2' \cap P_1$, and $l(O_l) = 0$.

Proof.

$$
\begin{aligned}
O_l &= (P_1' \cup P_2') \tilde{\cap} (-P_1) \\
&= \{(i, j), (j, i) \mid (i, j) \in P_1' \cup P_2' \text{ and } (j, i) \in -P_1\} \\
&= \{(i, j), (j, i) \mid (i, j) \in P_1' \cup P_2' \text{ and } (i, j) \in P_1\} \\
&= \{(i, j) \mid (i, j) \in P_1' \cup P_2' \cap P_1\} \\
&= P_1' \cup P_2' \cap P_1 .
\end{aligned}
$$

$l(O_l) = 0$ because the set O_l consists of cycles with zero length, each consisting of a pair of opposite arcs P_1 and $-P_1$. ■

Lemma (3)

The logical difference set between $P_1' \cup P_2'$ and P_1 is $(P_1' \cup P_2') - P_1 = P_1' \cup P_2' \cup (-P_1) \setminus O_l$.

Proof.

$$
\begin{aligned}
(P_1' \cup P_2') - P_1 &= \{(i, j) \mid (i, j) \in P_1' \cup P_2' \setminus (P_1' \cup P_2') \cap P_1\} \cup \\
&\qquad \{(j, i) \mid (i, j) \in P_1 \setminus (P_1' \cup P_2') \cap P_1\} \\
&= \{(i, j) \mid (i, j) \in P_1' \cup P_2' \cup (-P_1) \setminus (P_1' \cup P_2') \cap P_1\} \\
&= \{(i, j) \mid (i, j) \in P_1' \cup P_2' \cup (-P_1) \setminus O_l\} \\
&= P_1' \cup P_2' \cup (-P_1) \setminus O_l .
\end{aligned}
$$

■

Proposition (1)

The optimal set $\{P_1', P_2'\}$ has the smallest difference in length $Y = l(P_1') + l(P_2') - l(P_1) \geq 0$ from the shortest path P_1, among all the possible sets of arc-disjoint path pairs.

Proof.

We will prove this proposition by using the definition of the logical difference set.

With $l(-P_1) = -l(P_1)$, $(P_1' \cup P_2') - P_1 = P_1' \cup P_2' \cup (-P_1) \setminus O_l$,

we have
$$\begin{aligned} l((P_1' \cup P_2') - P_1) &= l(P_1' \cup P_2' \cup (-P_1) \setminus O_l) \\ &= l((P_1' \cup P_2') \cup (-P_1)) - l(O_l) \\ &= l(P_1' \cup P_2') + l(-P_1) \\ &= l(P_1') + l(P_2') + l(-P_1) \\ &= l(P_1') + l(P_2') - l(P_1) \\ &= Y. \end{aligned}$$
■

The following lemma shows that the logical difference set forms the shortest path in the modified network.

Lemma (4)

Given a directed network G (V, A) and pair (s, t) and let P_1 be the shortest path in this network. We define G (V, A′) as the network G (V, A) for which the path P_1 is replaced with $-P_1$. The logical difference set $(P_1' \cup P_2') - P_1$ between the optimal set of two arc-disjoint paths $\{P_1', P_2'\}$ and the shortest path P_1 forms the shortest path P_2 from node s to node t in G (V, A′).

Proof.

We will first prove that $P_2 = (P_1' \cup P_2') - P_1$ is a complete path from s to t in G (V, A′).

From Lemma (1), the optimal set of two arc-disjoint paths $P_1' \cup P_2'$ must contain either the first shortest path P_1 itself or some P_1 arcs on each of its two paths.

If $P_1' \cup P_2' \supset P_1$, without loss of generality, suppose $P_1' = P_1$, then $O_l = P_1 \cup (-P_1)$. With the definition of logical difference set, we have

$P_2 = ((P_1' \cup P_2') \cup (-P_1)) \setminus O_1 = (P_1 \cup P_2' \cup (-P_1)) \setminus (P_1 \cup (-P_1)) = P_2'$.

Hence P_2 must be a complete path from s to t.

If $P_1' \cup P_2'$ contains some P_1 arcs on each of its two paths, as $-P_1$ is the path from t to s in G (V, A′), and neither P_1' nor P_2' contains any $-P_1$ arcs, then the union $P_1' \cup P_2' \cup (-P_1)$ contains two cycles: one cycle consists of P_1' and $-P_1$, the other consists of P_2' and $-P_1$. When the set O_l is removed from the union set, the remaining arcs compose the logical difference set P_2. Hence P_2 must be a complete path from s to t.

Now we will prove that P_2 is the shortest path in G (V, A′). Assume that the shortest path in G (V, A′) is $P_3 \neq P_2$, then we must have $l(P_3) < l(P_2)$. As $l(P_2) = l(P_1') + l(P_2') - l(P_1)$ we have $l(P_3) < l(P_1') + l(P_2') - l(P_1)$ and $l(P_3) + l(P_1) < l(P_1') + l(P_2')$, which contradicts the assumption that $\{P_1', P_2'\}$ is the optimal set. ■

Lemma (5)

P_j is a shortest path from 1 to j in G and (i, j) is the last arc of P_j if and only if P_i which can be obtained by dropping (i, j) from P_j is a shortest path from 1 to i .

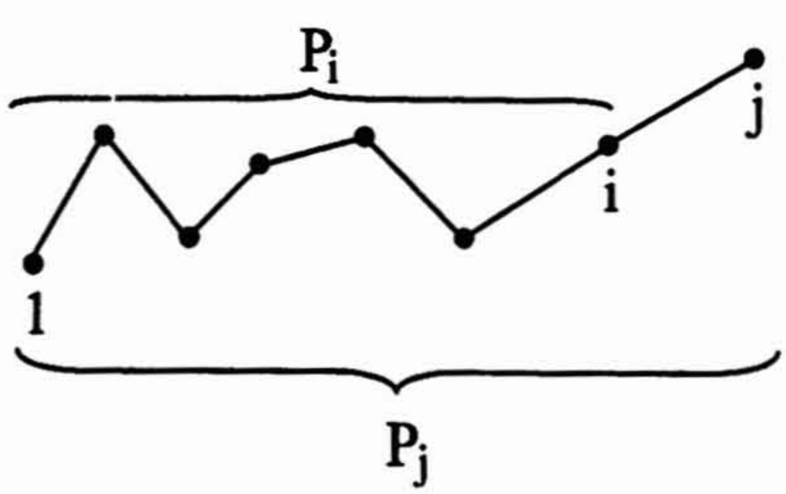

Fig. 4. Paths P_j and P_i

Proof.

Suppose that P_j is a shortest path from 1 to j. Assume P_i is not a shortest path from 1 to i. Then there is a path P_i^* that is shorter than P_i. Hence if we now add (i, j) to P_i^*, we get a path from 1 to j that is shorter than P_j. This contradicts our assumption that P_j is shorter. Therefore P_i is a shortest path from 1 to i.

Conversely, suppose that P_i is a shortest path from 1 to i. Obviously, P_j is a shortest path from 1 to j because P_j is the path which can be obtained by adding (i, j) to a shortest path P_i. ■

Many routing algorithms assume non-negative arc lengths to avoid a cycle of negative length appearing on a path. However, negative arc lengths introduced to a network in Bhandari's algorithm will not cause cycles in the routing process.

Theorem (1)

Given a directed network G (V, A) and source-destination pair (s, t) and let P_1 be the shortest path in this network. The modified network G (V, A′) is defined as the network G (V, A) for which P_1 is replaced with $-P_1$. A cycle containing some negative arcs in G (V, A′) will not have a negative length.

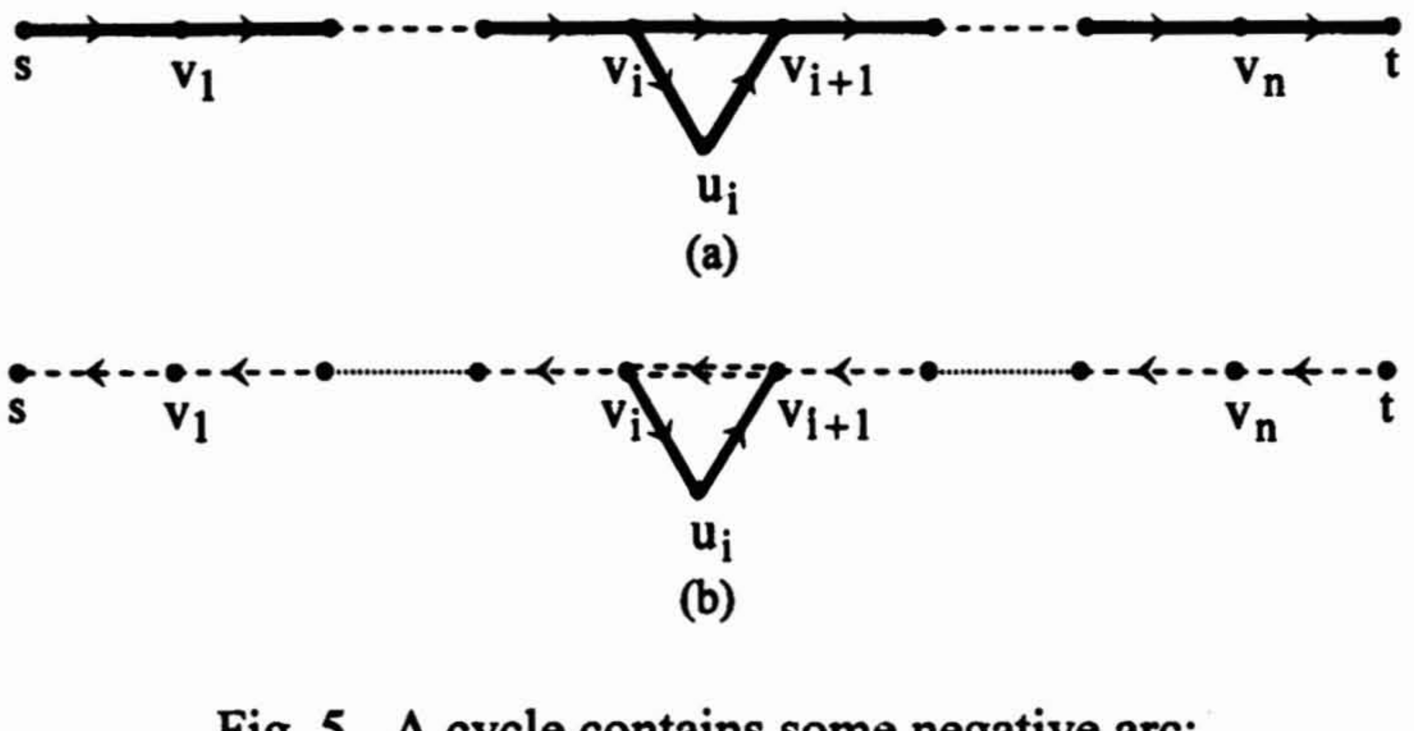

Fig. 5. A cycle contains some negative arc;

(a) The shortest path P_1 (s, t); and (b) A cycle containing some $-P_1$ arc

Proof.

Assume $s\,v_1 \ldots v_i\, v_{i+1} \ldots v_n\, t$ is the shortest path P_1 from node s to node t in G (V, A), as shown in Fig. 5 (a). The corresponding path $-P_1$ in G (V, A′) (Fig. 5 (b)) has an arc (v_{i+1}, v_i) which appears on cycle $B_1 = u_i\, v_{i+1}\, v_i\, u_i$.

Suppose the cycle B_1 has a negative length $l(B_1) = w_{u_i v_{i+1}} + w_{v_{i+1} v_i} + w_{v_i u_i} < 0$. Because $w_{v_{i+1} v_i} = -w_{v_i v_{i+1}}$, we must have $w_{u_i v_{i+1}} - w_{v_i v_{i+1}} + w_{v_i u_i} < 0$ and $w_{v_i u_i} + w_{u_i v_{i+1}} < w_{v_i v_{i+1}}$. Hence the sub-path $s\,v_1 \ldots v_i\, u_i\, v_{i+1}$ is shorter than the sub-path $s\,v_1 \ldots v_i\, v_{i+1}$.

This contradicts the assumption that $s\,v_1 \ldots v_i\, v_{i+1} \ldots v_n\, t$ is the shortest path. ■

The following theorem gives the optimality of the solution produced with Bhandari's algorithm.

Theorem (2)

Given a directed network G (V, A) and source-destination pair (s, t), the Bhandari's algorithm returns the optimal set for the APP problem.

Proof.

Let P_1 be the shortest path in the original network G (V, A) found in Step 1 of Bhandari's algorithm and P_2 be the shortest path in the modified network G (V, A′), found in Step 3 of Bhandari's algorithm. Let $\{P_1', P_2'\}$ be the solution set generated by Bhandari's algorithm. We will prove that $\{P_1', P_2'\}$ has the following properties.

(i) By construction of the solution set, we must have $P_1' \cap P_2' = \varnothing$. So the required path pair $\{P_1', P_2'\}$ is arc-disjoint.

(ii) Suppose the optimal set of arc-disjoint paths is $\{P_1'', P_2''\}$ instead of $\{P_1', P_2'\}$. According to Lemma (4), the logical difference set of $\{P_1'', P_2''\}$ with P_1 is the shortest path in the modified network G (V, A′). This contradicts that P_2 is the shortest path in modified network G (V, A′). So the solution set $\{P_1', P_2'\}$ generated by Bhandari's algorithm has minimal total length.

(iii) On Theorem (1), a cycle in the modified network G (V, A′) will not have a negative length.

By the above properties, the solution set returned by Bhandari's algorithm must be the optimal set. ■

Conclusion

In this paper Bhandari's algorithm which produces an optimal solution for (APP) problem has been described. We first showed that the optimal set for the (APP) problem is based on the shortest path. Secondly, we showed that the optimal set of two arc-disjoint path has the smallest difference in length from the shortest path among all the possible set of arc-disjoint paths. And we also proved that the logical difference set forms the shortest path in the modified network. Finally we proved that the solution produced with Bhandari's algorithm is optimal.

Acknowledgement

We are deeply indebted to Dr. Khin Maung Oo, Pro-rector of Lashio University for providing the full facilities and constant encouragement throughout the publication period.

We are also extremely grateful to Daw Oo Oo Myint, Professor, Head of Mathematics Department, Lashio University for her advise and support of this research work.

We profound gratitude also goes to Dr. Yi Myint, Professor, Head of Mathematics Department, West Yangon University and Dr. Nwe Wai Aung, Assistant Lecturer, Mathematics Department, Mandalay University, for their supervising and guideline.

References

Balakrishnan, V.K., (1991). *Introductory Discrete Mathematics*, Prentice Hall, Inc., New Jersey.

Bondy, J.A. and Murty, U.S.R., (1976). *Graph Theory with Applications*, Macmillan Press Ltd, London.

Guo, Y. and Kuipers, F. and Mieghem, P.V., (2003). Link-Disjoint Paths for Reliable QoS Routing, *Int. J. Commun. Syst.*, 16: 779 – 798 (DOI: 10. 1002/dac. 612).