On Edge Control Set of a Graph in Transportation Problems

Manoshi Kotoky
Department of Mathematics, Dibrugarh University, Dibrugarh : 786004, Assam, India
E-mail : mkotoky@yahoo.com
Arun Kumar Baruah
Department of Mathematics, Dibrugarh University, Dibrugarh : 786004, Assam, India
E-mail : baruah_arun123@rediffmail.com

ABSTRACT

One of the most significant problems in the analysis of the reliability of multi-state transportation systems is to find the minimal cut sets and minimal edge control sets. For that purpose there are several algorithms that use the minimal path and cut sets of such systems. In this paper we give an approach to determine the minimal edge control set. This approach directly finds all minimal edge control sets of a transport network. The main aim of the paper is to find optimal locations for sensors for detecting terrorists, weapons, or other dangerous materials on roads leading into major cities.

Keywords: **Edge Control Set, Minimal Edge Control Set, Sensors, Transport network.**

Date of submission: Aug 03, 2016 Date of Acceptance: Aug 18, 2016

1. INTRODUCTION

One of the most important and successful applications of quantitative analysis in solving business problems has been in the physical distribution of products, commonly referred to as transportation problems. Basically, the purpose is to minimize the cost of shipping goods from one location to another so that the needs of each arrival area are met and every shipping location operates within its capacity [1] [2]. Transportation problem is one of the subclasses of LPP's in which the objective is to transport various quantities of a single homogeneous commodity that are initially stored at various sources to different destinations in a way to minimize the total transportation cost, time, distance etc. These types of problems can be solved by general network methods. Transportation problems belong to a special class of network flow problems. Although these problems can be formulated as linear programming models, it is much more natural to formulate them in terms of nodes and arcs, taking advantage of the special structure of the problem. However, quantitative analysis has been used for many problems other than the physical distribution of goods. For example, it has been used to efficiently place employees at certain jobs within an organization, called an assignment problem [2].

Networks are essential components of our national infrastructure. Those networks could be used by terrorists seeking to attack dense urban populations with weapons of mass destruction. In particular, large urban road networks provide many routes that terrorists could use to get close enough to a major city to make a harmful attack. One approach envisioned for protecting urban areas from such attack is to deploy human-operated or fully automatic sensors on the roads, around cities to detect terrorists and their weapons so that they can be stopped before they come within range of their targets [3]. A key challenge to such an approach concerns how many sensors are to buy and where to locate them. Indeed, the size and density of road networks would seem to make the cost of buying and operating these sensors, prohibitive by requiring placement of sensors on hundreds if not thousands of road segments in order to protect any large city [2].

This challenge led to the work reported here, which shows that, the number of sensors required to cover every possible route into a city is not prohibitively large. We apply graph theory to find a minimal edge control set for a road network; i.e., to find a smallest set of road segments on which sensors must be placed to ensure that a terrorist traveling across the road network must encounter at least one sensor [4] [5] [6]. There are two situations occur when we use minimal edge control set to a connected network. For some case if we remove the minimal edge control set from the network, the remaining graph will be disconnected and for some cases it is connected. In this paper both the cases are discussed.

The work reported here specifically concerns finding optimal locations for sensors for detecting terrorists, weapons, or other dangerous materials on roads leading into major cities. However, this work is generally applicable to finding minimal edge control sets for any large network. It could be used to find optimal sensor locations on other transportation networks like railroads or subways. It could also be used to support offensive operations by locating a smallest set of segments in an adversary's network that would have to be cut in order to completely stop the flow through the network. Thus, the methodology presented here could have utility in other homeland security and military analysis.

2. Edge Control Set

To study the transportation problem, it has to be modeled mathematically by using a simple graph. The set of edges of the underlying graph will represent the communication link between the set of nodes. In the graph representing the transportation problem, the vertices will be joined by an edge if there is a communication link between the vertices otherwise not. In order to define an edge control set of a graph, we consider the underlying graph $G = (V, E)$ where $V(G)$ denotes the set of vertices of G and $E(G)$ denotes the set of edges of G. A cut-set F of G is called an edge control set of G if every flow of G is completely determined by F. A subset $F \subseteq E(G)$ is a cut-set of G if the removal of F from G disconnects G [1], [3]. Also it results in the increase in the number of components of G by one.

3. Minimal Edge Control Set

Let $G = (V, E)$ be a graph and $E(G)$ denotes the set of edges of G. An edge control set F is said to be minimal if any proper subset of F is not an edge control set of the graph G. As the edge control set of a graph is not unique, therefore it is important to find the set with the minimum number of edges.

Definition 3.1 Let $G = (V, E)$ be a graph, let H be a sub graph of G and $e \in E(H)$. We define

$C_H(e) = \{e\} \cup \{d \in E(H) : d$ is a cut edge of $H - \{e\}\}$

Then $C_H(e)$ is called the control of e in H.

Algorithm 3.2

Let G be a graph and a subset $F \subseteq E(G)$ is constructed by the following steps.
Step 1: Let $F := \emptyset$ and $H := G$
Step 2: While $E(H) \neq \emptyset$, select any edge $e \in E(H)$
$F := F \cup \{e\}$, $H := H - C_H(e)$.
Then F is the minimal edge control set of the graph G [7] [8].

Proof of the Algorithm

Let G be a graph. Then to prove that the set F constructed using the algorithm is the minimal edge control set. Let $F = \{e_1, e_2, e_3, \ldots e_t\}$ be the edges which are introduced to the set F in the same order as they are labeled and

$$E(G) = E(H) \sqsupseteq E(H_1) \sqsupseteq E(H_2) \ldots \sqsupseteq E(H_t) = \emptyset$$

be the sequence of sub graphs as they are generated using the algorithm.
As the removal of the set F disconnects the graph, therefore F is an edge control set of G and we are to show that F is the minimal edge control set of the graph G.
Let us suppose that there exist a set $F^I \subseteq F$, which is also an edge control set of G. Since $F^/ \subseteq F$, $\exists$ an edge $e_t \in F$ which is not in $F^/$. It implies that $e_t \in E(H_t)$, which is the smallest sub graph of the sequence of sub graph generated using the algorithm.
Since $e_t \notin F^/$, then there exists an edge e_t of the sub graph $E(H_t)$ which is connected to somevertices of the graph G and the removal of the set will not disconnect the graph G. Hence,

$$E(G) = E(H) \supset E(H_1) \supset E(H_2) \ldots \supset E(H_t) \neq \emptyset$$

This implies that there exists at least one edge of G which is connected to some vertices of the graph G. Therefore $F^/$ cannot be an edge control set of G which is a contradiction to`$F^/ \subseteq F$. Hence F is a minimal edge control set of G constructed by the algorithm.

4. EXAMPLES

(i) Disconnected Case:

Let us consider a transportation problem with road segments as shown in **Fig. 1.** Here nodes represent the different places of a city and edges represent the roads joining them. The corresponding graphs are shown in the figures below:

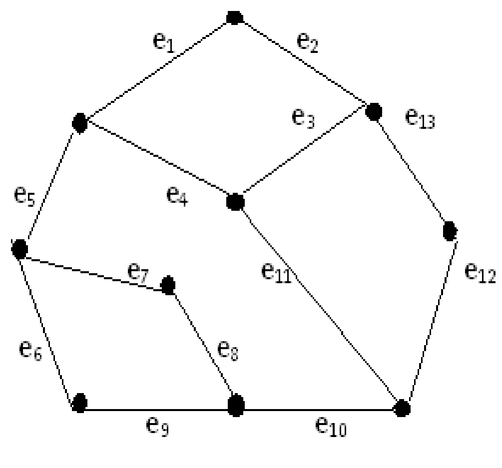

Fig. 1: A graph with thirteen road segments

To start with the Algorithm (3.2), we consider $F := \emptyset$ and $H := G$. Now we select any edge e_1 such that $e_1 \in E(H)$ i.e. $E(H) \neq \emptyset$. Thus we have,

$C_H(e_1) = \{e_1\} \cup \{e_2\}$

$= \{e_1, e_2\}$
$F = F \cup \{e_1\} = \{e_1\}$
$H: = H - C_H(e_1)$
$= \{e_3, e_4, e5, e_6, e_7, e_8, e_9, e_{10}, e_{11}, e_{12}, e_{13}\}$

Where the first subgraph ***H:*** is as shown in **Fig. 2** below:

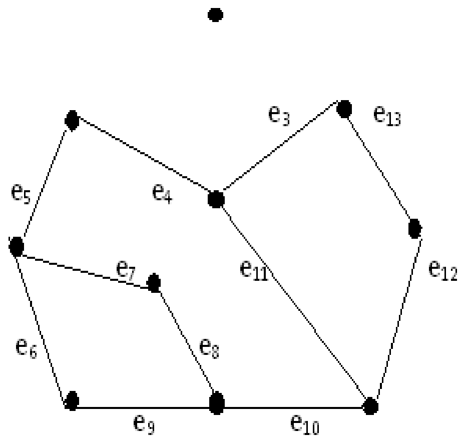

Fig. 2 : H is the first subgraph obtained applying the Algorithm (3.2)

Again since $E(H) \neq \emptyset$, let us select any edge $e_5 \in E(H)$. Then we obtain

$C_H(e_5) = \{e_5\} \cup \{e_4\}$
$= \{e_4, e_5\}$
$F = \{e_1\} \cup \{e_5\} = \{e_1, e_5\}$
$H: = H - C_H(e_5)$
$= \{e_3, e_6, e_7, e_8, e_9, e_{10}, e_{11}, e_{12}, e_{13}\}$

Where the second subgraph ***H:*** is as shown in **Fig. 3** below:

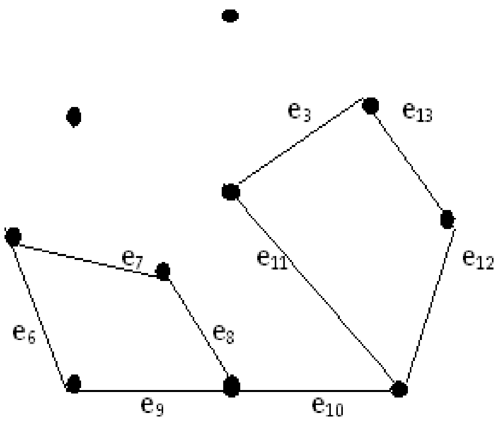

Fig. 3 : H is the second subgraph obtained applying the Algorithm (3.2)

Since $E(H) \neq \emptyset$, let us select any edge $e_{10} \in E(H)$. Then we obtain

$C_H(e_{10}) = \{e_{10}\} \cup \{e_8, e_9, e_{11}, e_{12}\}$
$= \{e_8, e_9, e_{10}, e_{11}, e_{12}\}$
$F = \{e_1, e_5\} \cup \{e_{10}\} = \{e_1, e_5, e_{10}\}$
$H: = H - C_H(e_{10})$
$= \{e_3, e_6, e_7, e_{13}\}$

Where the third subgraph ***H:*** is as shown in **Fig. 4** below:

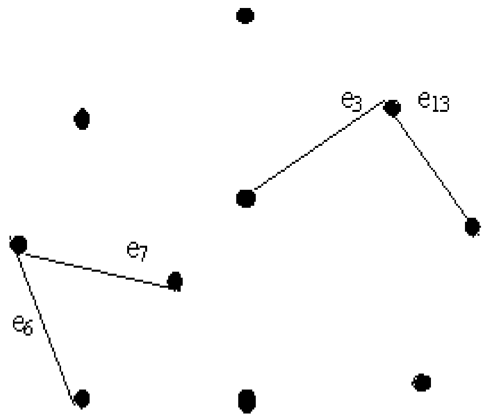

Fig. 4: H is the third subgraph obtained applying the Algorithm (3.2)

Now, let $e_6 \in E(H)$, then we obtain

$C_H(e_6) = \{e_6\} \cup \{e_7\}$
$= \{e_6, e_7\}$
$F = \{e_1, e_5, e_{10}\} \cup \{e_6\} = \{e_1, e_5, e_6, e_{10}\}$
$H: = H - C_H(e_6)$
$= \{e_3, e_{13}\}$

Where the fourth subgraph ***H:*** is as shown in **Fig. 5** below:

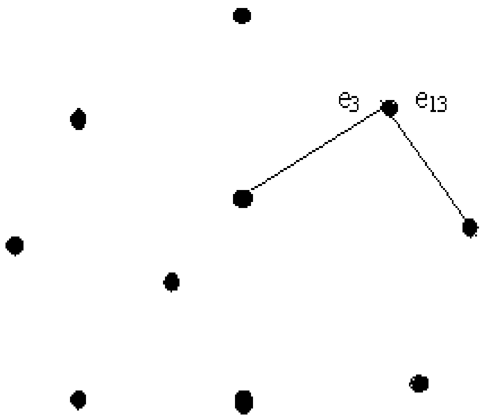

Fig. 5 : H is the fourth subgraph obtained applying the Algorithm (3.2)

Int. J. Advanced Networking and Applications
Volume: 08 Issue: 01 Pages: 3003-3008 (2016) ISSN: 0975-0290

Finally, let $e_3 \in E(H)$, then we obtain $C_H(e_3)=\{e_3, e_{13}\}$ and $F=\{e_1, e_3, e_5, e_6, e_{10}\}$. The new ***H:*** consists of all isolated vertices, i.e., $E(H)=\emptyset$. The fifth subgraph with all isolated vertices is shown in **Fig. 6**

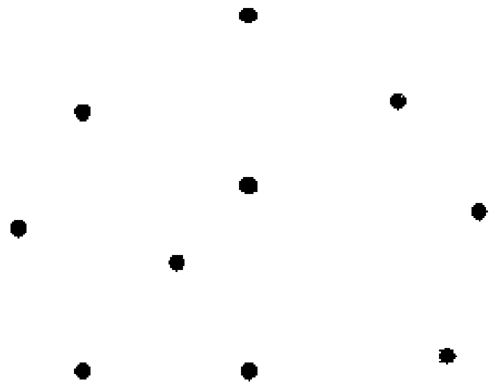

Fig. 6 : H is the fifth and final subgraph obtained applying the Algorithm (3.2)

Therefore $F=\{e_1, e_3, e_5, e_6, e_{10}\}$ is a minimal edge control set of ***G.*** The graph obtained after removing ***F*** from ***G*** is as shown in **Fig.7** below:

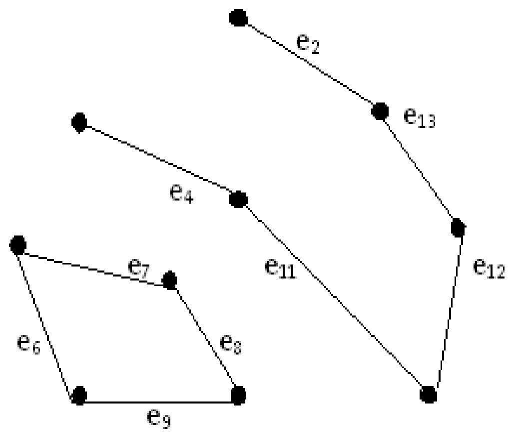

Fig. 7: Graph obtained after removing minimal edge control set

Finally, it is also clear from the above diagram that the above graph is disconnected having two components and no other set having less number of edges than ***F*** disconnect the graph, therefore ***F*** is a minimal edge control set of the graph ***G*** constructed using the algorithm. From the above discussion it is clear that sensors have to be placed in the edges e_1, e_3, e_5, e_6 and e_{10} which will provide complete information about the whole transportation network.

(ii) Connected Case:

Let us consider a transportation problem with road segments as shown in **Fig. 8.** Here nodes represent the different places of a city and edges represent the roads joining them. The corresponding graphs are shown in figures below:

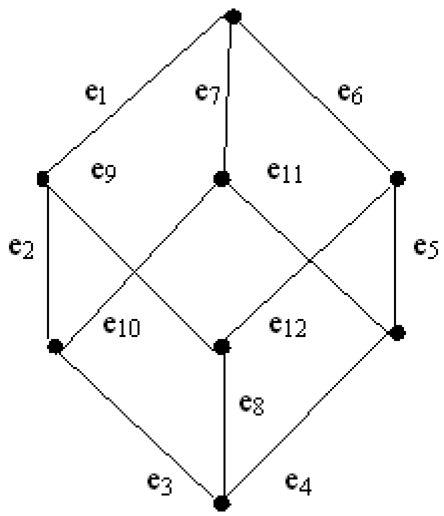

Fig. 8: A graph with twelve road segments

To start with the Algorithm (3.2), we consider $F:=\emptyset$ and $H:=G$. Now we select any edge e_1 such that $e_1 \in E(H)$ i.e. $E(H) \neq \emptyset$. Thus we have,

$$C_H(e_1)=\{e_1\} \cup \{e_6, e_7\}$$
$$=\{e_1, e_6, e_8\}$$
$$F=F \cup \{e_1\}=\{e_1\}$$
$$H:=H-C_H(e_1)$$
$$=\{e_2, e_3, e_4, e_5, e_6, e_8, e_9, e_{10}, e_{11}, e_{12}\}$$

Where the first subgraph ***H:*** is as shown in **Fig. 9** below:

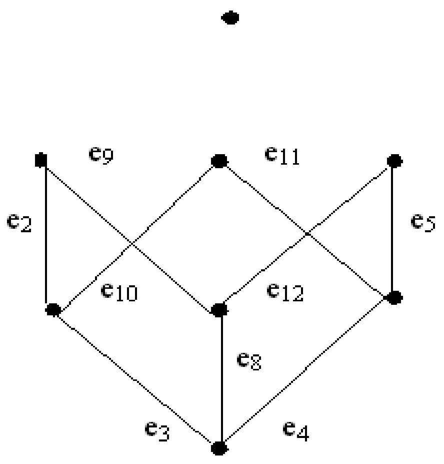

Fig. 9 : H is the first subgraph obtained applying the Algorithm (3.2)

Let $e_2 \in E(H)$, then we obtain

$C_H(e_2) = \{e_2\} \cup \{e_9\}$
$= \{e_2, e_9\}$
$F = \{e_1\} \cup \{e_2\} = \{e_1, e_2\}$
$H: = H - C_H(e_2)$
$= \{e_3, e_4, e, e_8, e_{10}, e_{11}, e_{12}\}$

Where the second subgraph ***H:*** is as shown in **Fig. 10** below:

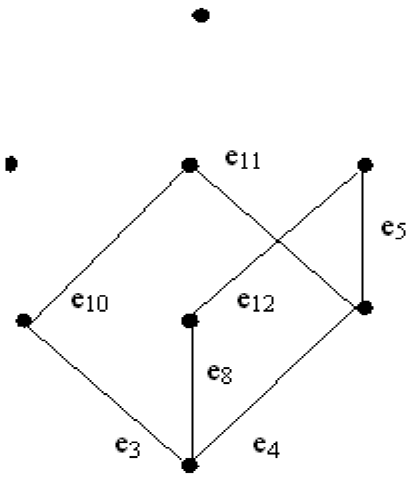

Fig. 10: H is the second subgraph obtained applying the Algorithm (3.2)

Let $e_3 \in E(H)$, then we obtain
$C_H(e_3) = \{e_3\} \cup \{e_{10}\}$
$= \{e_3, e_{10}\}$
$F = \{e_1, e_2\} \cup \{e_3\} = \{e_1, e_2, e_3\}$
$H: = H - C_H(e_3)$
$= \{e_4, e_5, e_8, e_{11}, e_{12}\}$

Where the third subgraph ***H:*** is as shown in **Fig. 11** below:

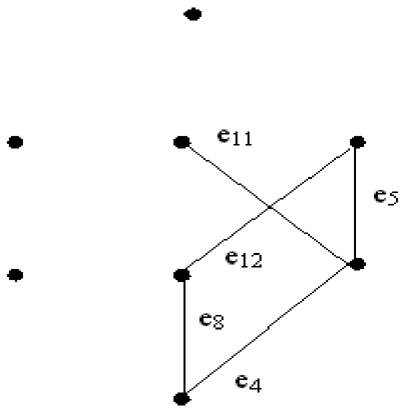

Fig. 11 : H is the third subgraph obtained applying the Algorithm (4.2)

Let $e_4 \in E(H)$, then we obtain
$C_H(e_4) = \{e_4\} \cup \{e_8\}$
$= \{e_4, e_8\}$
$F = \{e_1, e_2, e_3\} \cup \{e_4\} = \{e_1, e_2, e_3, e_4\}$
$H: = H - C_H(e_4)$
$= \{e_5, e_{11}, e_{12}\}$

Where the fourth subgraph ***H:*** is as shown in **Fig.12** below:

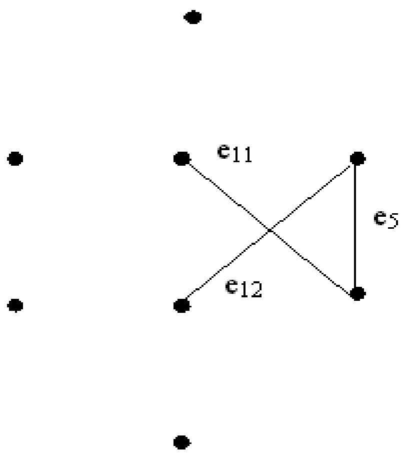

Fig. 12 : H is the fourth subgraph obtained applying the Algorithm (3.2)

Let $e_5 \in E(H)$, then we obtain
$C_H(e_5) = \{e_5\} \cup \{e_{11}, e_{12}\}$
$= \{e_5, e_{11}, e_{12}\}$
$F = \{e_1, e_2, e_3\} \cup \{e_5\} = \{e_1, e_2, e_3, e_4, e_5\}$
$H: = \emptyset$

The third subgraph with all isolated vertices is shown in **Fig. 13**

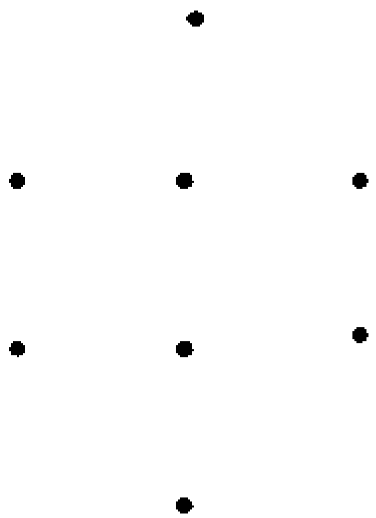

Fig. 13 : H is the fifth and final subgraph obtained applying the Algorithm (3.2)

Therefore ***F= { e_1,e_2 e_3, e_4, e_5 }*** is a minimal edge control set of ***G.*** The graph obtained after removing ***F*** from ***G*** is as shown in **Fig. 14** below:

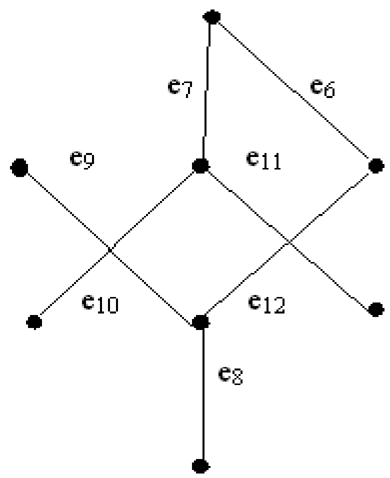

Fig. 14: Graph obtained after removing minimal edge control set

Finally, it is also clear from the above diagram that ***F*** is a minimal edge control set of the graph ***G*** constructed using the algorithm. The graph remains connected even after removal of ***F***. From the above discussion it is clear that sensors have to be placed in the edges e_1, e_2, e_3, e_4 and e_5 which will provide complete information about the whole transportation network.

5. CONCLUSION

In this paper we have used minimal edge control set as a graph theoretic tool to study the transportation problem. Minimal edge control set determines the whole transportation flow. This can be achieved by placing traffic sensors on each of the minimal edge control set of the transportation network which will provide complete information of the transport network. Here two examples are considered to explain the use of minimal edge control set. In one case minimal edge control set is used as a cut set whose removal disconnects the graph. If terrorist wanted to attack a major city, they can be stopped by removing these edges. In the other example we consider another minimal edge control set whose removal does not disconnect the network. The sensors can be placed on each edge control set and from the definition of an edge control set these sensors will provide complete information for the control system. Thus optional locations for the traffic sensors can be obtained by using edge control set.

6. Acknowledgements

The authors would like to thank Professor Arun Kumar Baruah for helpful suggestions which led to a major revision of the paper.

REFERENCES

[1] D. B. West, *Introduction to Graph Theory* (2nd Edition Prentice Hall Inc., New Jersey, 2001).

[2] R.K. Ahuja , T.L. Magnanti, and J.B. Orlin. "Maximum Flows: Basic Ideas." Chapter 6 of *Network Flows Theory, Algorithms, and Applications*.Prentice Hall, 1993.

[3] R. J. Wilson, J. J. Watkins, *Graph : An Introductory Approach* (John Wiley & Sons, New York, 1989).

[4] V.K. Balakrishnan "Graph Theory." Appendix of *CombinatoricsIncluding Concepts of Graph Theory*, Schaum's Outline Series.McGraw-Hill, 1995.

[5] G.H. Bradley , G.G. Brown, and G.W. Graves. "Design and Implementation of Large-Scale Primal Transshipment Algorithms." *Management Science* 24, no. 1 (1977): 1-34.

[6] M.O. Ball "Design of Survivable Networks." Chapter 10 of *Handbooks in OR & MS, Vol. 7*, *Network Models*, M.O. Ball et al., Eds. Elsevier Science B.V., 1995.

[7] W. Gu, X. Jia, On a Traffic Sensing Problem, *Preprint, Texas State University*, 2006.

[8] W. Gu, X. Jia, On a Traffic Control Problem, *8th International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN' 05)*, 2005.