*Journal of the Korean Statistical Society (2000),* **29**: *3, pp 297–306*

# An Efficient Method on Constructing $k$-Minimal Path Sets for Flow Network Reliability†

**Seung Min Lee[1] and Dong Ho Park[1]**

ABSTRACT

An efficient method of constructing $k$-minimal path sets to evaluate the reliability of a flow network is presented. The network is considered to be in a functioning state if it can transmit a maximum flow which is greater than or equal to a specified amount of flow, $k$ say, and a $k$-minimal path set is a minimal set of branches that satisfies the given flow constraint. In this paper, under the assumption that minimal path sets of the network are known, we generate composite paths by adding only a minimal set of branches at each iteration to get $k$-minimal path sets after possibly the fewest composition, and compute maximum flow of composite paths using only minimal path sets. Thereby we greatly reduce the possible occurrence of redundant composite paths throughout the process and efficiently compute the maximum flow of composite paths gernerated. Numerical examples illustrate the method.

*Keywords:* Flow network; Capacity; Maximum flow; Composite Path; $k$-minimal path set

## 1. INTRODUCTION

Evaluation of capacity related reliability of a flow network has attracted considerable attention in the literature. The network is represented by a probabilistic graph $G(V, E)$, which consists of a set $V$ of nodes and a set $E$ of branches (edges). Each branch may have different flow capacity and the network may be required to transmit a specified amount of flow from the source (input) node to the terminal

---

†This work was supported by the Hallym University research fund, 1997.

[1]Department of Statistics, Hallym University, Chunchon 200-702, Korea.

(output) node. Some examples of such networks are a computer communication network which allows only a fixed amount of data exchange among different terminals of various computer centers, a transport system of a large town which limits maximum traffic on various roads, and a hydraulic system which carries gas or fluid through a pipeline network with capacity limitations. In these cases, the performance of network is naturally determined by the amount of flow that can be transmitted from the source node to the terminal node and successful operation of network is not necessarily characterized by connectivity only, but by the maximum flow that can be transmitted through the network. The network reliability is measured as the probability of successfully transmitting the required amount of flow from the source node to the terminal node.

A number of algorithms have been proposed to obtain the above capacity related reliability. The method suggested by Lee (1980) is based on the concept of lexicographic ordering and a labeling scheme is used to route the flow through the network. Another approach given in Qiu and Zhong (1994) use the initial valid group to generate all valid groups by successive replacement of branches one by one. All the resulting valid groups are mutually disjoint and used to evaluate the network reliability. Under the assumption that all minimal path sets of the network are known, Misra and Prasad (1982) propose a method utilizing a failure path list to enumerate the composite paths, but Rai and Soh (1991) present a counter example to show that it does not generate enough composite paths and fails to give correct results in general. Aggarwal et al. (1982) and Aggarwal (1988) also discuss the similar methods based on the composite path enumeration approach, but both have some drawbacks in computing maximum flow of composite paths. Varshney (1994) expands the method of Aggarwal (1988) to allow the branches having multiple states of capacities with corresponding probabilities, and Schanzer (1995) argues with counter examples that it also fails in certain cases. Rai and Soh (1991) try to complement and correct the drawbacks of the preceeding results. The method generates composite paths categorized by the number of minimal path sets involved in composition and uses minimal cut sets to compute maximum flow. Hence, the method may generate a large number of redundant composite paths and needs to find minimal cut sets from the given minimal path sets.

Our method generates composite paths by adding, each time, only a minimal set of branches to satisfy the network flow requirement after possibly the fewest composition, and compute maximum flow of composite paths using only minimal path sets. Thereby we greatly reduce the possible occurrence of redundancy in terms of duplication and absorption throughout the process and save time and efforts in computing maximum flow of composite paths. Section 2 presents a simple method of computing maximum flow of a composite path using flow augmenting paths, each of which is a minimal path set and illustrates the concept of $k$-minimal path sets. Section 3 gives the descriptions on the methodology and algorithm to construct all $k$-minimal path sets. Once the $k$-minimal path sets are found, the exact evaluation of flow network reliability is straightforward. The upper and lower bounds for the reliabilities based on the $k$-minimal path (cut) sets are also discussed briefly. In Section 4, numerical examples are presented to illustrate our methods.

**Notation**

$C$: set of branches $= \{1, 2, \cdots, n\}$

$c_i$: flow capacity of branch $i$

$\hat{c}$ : capacity vector $= (c_1, \cdots, c_n)$

$X_i$: random variable indicating the state of branch $i$,

$$X_i = \begin{cases} 1 & \text{if branch } i \text{ is functioning} \\ 0 & \text{if branch } i \text{ is failed} \end{cases}$$

$\hat{X}$: random state vector $= (X_1, \cdots, X_n)$

$\hat{x}$ : binary vector values that $\hat{X}$ can assume

$p_i$: $P\{X_i = 1\}$

$M(\hat{x} : \hat{c})$: performance of the network when $\hat{X} = \hat{x}$, defined as the maximum amount of flow that can be transmitted from the source node to the terminal node

$M(B) = M((\hat{1}^B, \hat{0}) : \hat{c})$: maximum flow when branches only in $B(\subset C)$ function

## 2. MAXIMUM FLOW OF A COMPOSITE PATH

**Assumptions**

1. The nodes are perfect and each has no capacity limit.
2. The branches are independent and either function or fail with known probabilities.
3. All the branches are directed and each branch flow is bounded by the capacity of the branch.

4. No information or flow can be transmitted through a failed branch.
5. The network is good i.f.f. a specified amount of flow can be transmitted from the source node to the terminal node.

**Nomenclatures**

A vetor $\hat{x}$ is said to be a *k-minimal path vector* if $M(\hat{x} : \hat{c}) \geq k$ and $M(\hat{y} : \hat{c}) < k$ for all $\hat{y} < \hat{x}$. For a $k$-minimal path vector $\hat{x}$, the set $A = \{i|x_i = 1\}$ is called *k-minimal path set* ($k$-mps). A vector $\hat{x}$ is said to be a *k-minimal cut vector* if $M(\hat{x} : \hat{c}) < k$ and $M(\hat{y} : \hat{c}) \geq k$ for all $\hat{y} > \hat{x}$. For a $k$-minimal cut vector $\hat{x}$, the set $K = \{i|x_i = 0\}$ is called a *k-minimal cut set* ($k$-mcs).

We assume that all the branches are directed without loss of generality, since an undirected branch can be replaced by two oppositely directed branches of the same capacity as that of the undirected branch. A minimal path set (mps) of a directed network is a minimal set of branches which, viewed as a path, connects source node to terminal node. An mps, as a path, is not necessarily a directed path. $M(\hat{x} : \hat{c})$ measures the maximum amount of flow that can be passed through the network successfully, given $\hat{x}$, and $\hat{c}$. $M(\hat{x} : \hat{c})$ is non-decreasing in each argument, and $M((\hat{1}^A, \hat{0}) : \hat{c}) = min_{i \in A}\{c_i\}$ for a 1-mps $A$ of the network. Note that an mps is a 1-mps if and only if it is a directed path.

Let $k, k \geq 1$, be a pre-specified amount of flow. Then, a $k$-mps is a minimal set of branches which ensures successful operation of the network. The network reliability evaluation requires four major procedures:

(1) determination of all mps's of the network,

(2) generation of enough composite paths to find all possible $k$-mps's,

(3) computation of maximum flow for each composite path generated,

(4) transformation of the knowledge on $k$-mps's into reliability expression.

However, (1) and (4) have been extensively covered in the literature. See, for example, Aggarwal et al. (1982) and the references therein. Thus, we only discuss (2) and (3) in details. The procedure (3) is essentially the same as computing maximum flow of a network and can be computed by applying well known methods such as Ford-Fulkerson algorithm. The method searches for a flow augmenting path, and then increase the flow along this path. See Ford and Fulkerson (1962) for details. But it is slow and difficult to computerize.

In this section, we suggest a simple method of computing maximum flow when all mps's of the network are known. Let $\hat{f}$ be a feasible flow pattern and let $f_{st}$ be the corressponding flow value from $s$ to $t$ with respect to $\hat{f}$. An mps $A$ is said to be a *flow augmenting* mps with respect to $\hat{f}$ if all forward branches of $A$ are not saturated, and all reverse branches are not flowless. Here, a branch $i$ of $A$ is said to be a *reverse branch* if in traversing from $s$ to $t$ on $A$ the branch $i$ is directed reversely to the given direction of $i$ in the network. Otherwise, it is a *forward branch* of $A$. We say that a branch $i$ is *saturated* with respect to $\hat{f}$ if $f_i = c_i$ and is *flowless* if $f_i = 0$. From the properties of flow augmenting paths, we obtain the following lemma.

**Lemma 2.1.** For the shortest flow augmenting mps $A$ with respect to $\hat{f}$,

$$f'_{st} = f_{st} + f(A)$$

where $f(A) = min\{min_{i \in F_A}(c_i - f_i), min_{i \in R_A}(f_i)\}$, $f'_i = f_i + f(A)$ for $i \in F_A$, $f'_i = f_i - f(A)$ for $i \in R_A$, and $F_A(R_A)$ is the set of forward (reverse) branches of $A$.

By applying Lemma 2.1 repeatedly, we can easily compute $M(\hat{x} : \hat{c})$. Let $B = \{i | x_i = 1\}$ and rewrite $\hat{x} = (\hat{1}^B, \hat{0})$, so that $M(\hat{x} : \hat{c}) = M((\hat{1}^B, \hat{0}) : \hat{c}) = M(B)$. Then, starting with the zero flow pattern $\hat{f} = \hat{0}$, we find the shortest flow augmenting mps $A$ with respect to $\hat{f}$, which is a subset of $B$. If there is no such mps, then $M(B) = 0$. By applying Lemma 2.1, we have $f'_{st} = f_{st} + f(A)$ where $f_{st} = 0$. We increase the flow of the network by repeating this process with $f_{st} = f(A)$ and $\hat{f} = \hat{f}'$ until there is no more flow augmenting mps with respect to $\hat{f}$, at which moment we have $M(B) = f_{st}$.

In the followings, we discuss some properties of a $k$-mps. Let $A$ be a $k$-mps. Then it is obvious that $A$ is a $k'$-mps whenever $k \leq k' \leq M(A)$, since $M(A - \{i\}) < k'$ for $i \in A$. Here, $A - \{i\}$ is the set obtained from $A$ by deleting $i$. Another useful property is that any $k$-mps, $k \geq 1$, can be expressed as a union of 1-mps's. This is true since for each $i \in A$, it is always possible to find a 1-mps containing $i$, say $A_{(i)}$, such that $A_{(i)} \subset A$ and thus $A = \cup_{i \in A} A_{(i)}$. We note that the level $k$ of an mps does not have to be integer-valued. The level "1" is used just as a separating point between no flow (network completely failed), and some flow (allowing partial functioning of the network). In general, if we let $m = min\{c_i | i \in C\}$, then for $0 < \alpha \leq m$, $\alpha$-mps's are the usual 1-mps's. Our main point is that we can construct any $k$-mps from the usual 1-mps's. We also note that there is no mps of level $k$ for $k$ exceeding $M(C)$, which is the maximum amount of flow possible when all branches of the network function.

## 3. $k$-MINIMAL PATH SETS AND RELIABILITY

A 1-mps of the flow network is also a $k$-mps for $k, 1 \le k \le M(A)$. However, since a $k$-mps can be represented as a union of 1-mps's, there is still a possibility of constructing a $k$-mps by using 1-mps's even when their maximum flow is less than $k$. For example, let $A_1$ and $A_2$ be two disjoint 1-mps's such that $M(A_1) = k_1 < k$ and $M(A_2) = k_2 < k$. Then, $A_1$ and $A_2$ are not $k$-mps's but $A_1 \cup A_2$ is a $k$-mps if it contains no other 1-mps as a subset and if $max(k_1, k_2) < k \le k_1 + k_2$. This suggests that we can find $k$-mps's as unions, which are called *composite paths*, of different combinations of 1-mps's whose maximum flow is less than $k$.

This section describes an efficient algorithm on how to find all $k$-mps's of the network without producing much redundancy throughout the process. Finding a $k$-minimal path set is essentially adding or subtracting some branches and, each time, checking the maximum flow of resulting set of branches against the flow constraint. Our basic idea is to add, each time, a choice of minimal set of branches which gives maximal increase on maximum flow and such a choice is made from the set of failure 1-mps's. In algorithm, we partition the set of all 1-mps's into two parts at each iteration; the current working sequence $B$ and the corresponding set of available 1-mps's, FMPS say, each of which is not a subset of $B$. Initially we set $B = \emptyset$, and FMPS={all 1-mps's}. With current $B$ and FMPS, we find a suitable choice $A$ and construct a composite path $B \cup A$. A 1-mps $A \in$FMPS is said to be a *suitable choice*, if $A$ gives the largest increase on maximum flow among all 1-mps's in FMPS$'$($\subset$FMPS) where FMPS$'$ = $\{A' |$ there is no $A'' \in$FMPS such that $(A'' - B)$ is a proper subset of $(A' - B)\}$. Each time a composition is made, the resulting composite path is tested against the requirement $k$ using the method described in Section 2. If it satisfies the requirement, report it as a candidate for $k$-mps and repeat the process with $B$ and FMPS=FMPS$-\{A\}$. If not, repeat the process with $B = B \cup A$ and FMPS=FMPS$-\{A\}$. If there is no choice, then change the choice added last on $B$ and repeat the process with the corresponding FMPS. We also check $M(ALL)$ to make sure that some $k$-mps's are possible with current $B$, where $ALL$ is the union of $B$ and all 1-mps's in the corresponding FMPS. We proceed only when $M(ALL) \ge k$ and find all $k$-mps's starting with $B$. The search for the $k$-mps's terminates when it is found that no more candidate for $k$-mps possible with $B = \emptyset$. The set of $k$-mps's is obtained by removing the possible redundancy in the final list of candidates at the completion of the process.

**Algorithm:**

To find all $k$-mps's, given all 1-mps's of the network,

1. Initially, set $m = 0$, $B_0 = \emptyset$, and $\text{FMPS}_0$={all 1-mps's}.
   If $M(ALL) < k$, then $STOP$, else go to 3.
2. Set $\text{FMPS}_m$=$\text{FMPS}_m - \{A \in \text{FMPS}_m | A \subset B_m\}$.
   If $\text{FMPS}_m = \emptyset$, then go to 5.
3. With $B_m$ and $\text{FMPS}_m$, find a suitable choice $A$.
   Set $\text{FMPS}_m$=$\text{FMPS}_m - \{A\}$.
   If $M(B_m) + M(A) < k$, then go to 4.
   If $M(B_m \cup A) \geq k$, then report $B_m \cup A$ as a candidate for a $k$-mps and go to 3.
4. If $\text{FMPS}_m = \emptyset$, then go to 5.
   Set $B_{m+1} = B_m \cup A$, $\text{FMPS}_{m+1}$=$\text{FMPS}_m$, $m = m + 1$, and go to 2.
5. Set $m = m - 1$.
   If $m < 0$, then $STOP$.
   If $M(ALL) < k$, then go to 5, else go to 3.

Given the $k$-mps's, the $k$-mcs's can be generated by an inversion technique described in Locks (1978). Once the $k$-mps's and $k$-mcs's are given, the upper and lower bounds for the flow network reliability can be readily constructed by the well-known methods described in Barlow and Proschan (1982). We reproduce these bounds here for completeness. Let $A_1, \cdots, A_p$ be the minimal path set and $K_1, \cdots, K_k$ be the minimal cut set of a coherent binary system. Then the path-cut bounds and min-max bounds for the system reliability are given as

$$\left( \prod_{j=1}^{k} \coprod_{i \in K_j} p_i, \coprod_{j=1}^{p} \prod_{i \in A_j} p_i \right)$$

and

$$\left( \max_{1 \leq r \leq p} \prod_{i \in A_r} p_i, \min_{1 \leq s \leq k} \coprod_{i \in K_s} p_i \right),$$

respectively. These bounds are computed for the given network in the next section.

## 4. NUMERICAL EXAMPLES

**Example 4.1.** The bridge network shown in Figure 4.1 has the following flow capacity: $c_1 = 6, c_2 = 2, c_3 = 1, c_4 = 3, c_5 = 2$.

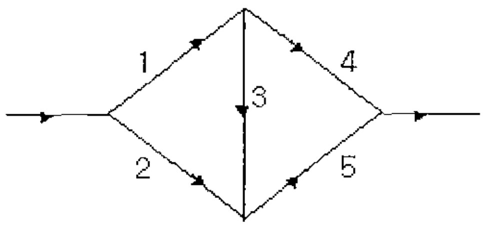


Figure 4.1: A Bridge Network

Firstly, we obtain all the 1-mps's as $A_1 = \{1,4\}, A_2 = \{2,5\}, A_3 = \{1,3,5\}$. To determine the 2-mps's , we check first if $M(A_j) \geq 2, j = 1,2,3$. Since $M(A_1) = 3$ and $M(A_2) = 2$, both $A_1$ and $A_2$ are the 2-mps's. However, $M(A_3) = 1$ and thus $A_3$ is not a 2-mps's. Since there is no candidates to generate more 2-mps, the procedure ends and there are two 2-mps's $A_1$ and $A_2$. For 4-mps's, none of $A_1, A_2$ and $A_3$ is a 4-mps. Constructing composite paths, we have two 4-mps's $A_1 \cup A_2 = \{1,2,4,5\}$ and $A_1 \cup A_3 = \{1,3,4,5\}$. Note that $A_2 \cup A_3$ is not a 4-mps, since $M(A_2) + M(A_3) < 4$. In the same manner, we generate $k$-mps's for $k$=2,3,4,5, as in Table 4.1. The table also lists the $k$-mcs's and the exact network reliability when $P_i = 0.8$, $i = 1, \cdots, 5$. For example, for $k$=4, we have the reliability $R_4 = P\{1,2,4,5\} + P\{1,3,4,5\} - P\{1,2,3,4,5\} = (0.8)^4 + (0.8)^4 - (0.8)^5 = 0.49152$ by using two 4-mps's and the fact that the branches are independent. It is also straightforward to compute the upper and lower bounds for the network reliability by the path-cut methods and the min-max methods which are given in Table 4.2 when $p_i = 0.8$, $i = 1, \cdots, 5$.

Table 4.1: $k$-mps, $k$-mcs and exact reliability of the bridge network

| $k$ | $k$-mps | $k$-mcs | network reliability |
|---|---|---|---|
| 5 | {1,2,4,5} | {1}{2}{3}{4} | 0.4096 |
| 4 | {1,2,4,5}{1,3,4,5} | {1}{4}{5}{2,3} | 0.4915 |
| 3 | {1,4} | {1}{4} | 0.6400 |
| 2 | {1,4}{2,5} | {1,2}{1,5}{2,4}{2,5} | 0.8704 |
| 1 | {1,4}{2,5}{1,3,5} | {1,2}{1,5}{4,5}{2,3,4} | 0.8909 |

Table 4.2: Reliability bounds for the bridge network

| $k$ | path-cut bounds | min-max bounds |
|---|---|---|
| 5 | (0.4096, 0.4096) | (0.4096, 0.8000) |
| 4 | (0.4915, 0.6514) | (0.4096, 0.8000) |
| 3 | (0.6400, 0.6400) | (0.6400, 0.8000) |
| 2 | (0.8493, 0.8704) | (0.6400, 0.9600) |
| 1 | (0.8777, 0.9367) | (0.6400, 0.9600) |

**Example 4.2.** A network shown in Figure 4.2 has 7 directed branches with the flow capacity $c_1 = 6, c_2 = 2, c_3 = 1, c_4 = 3, c_5 = 2, c_6 = 2, c_7 = 3$.

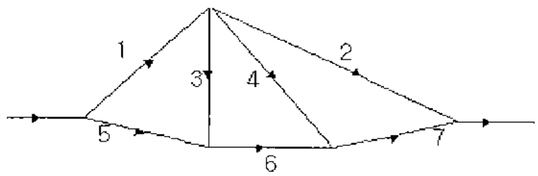


Figure 4.2: A 7-Branch Network

For illustrative purpose, we apply our algorithm to find the 3-mps's. The 1-mps's are obtained as $A_1 = \{1,2\}, A_2 = \{5,6,7\}, A_3 = \{1,4,7\}, A_4 = \{1,3,6,7\}$. Since $M(A_3) = 3$, $A_3$ is a 3-mps. Constructing composite paths, $A_1 \cup A_2 = \{1,2,5,6,7\}$ and $A_1 \cup A_4 = \{1,2,3,6,7\}$ are the 3-mps's. Note that $A_2 \cup A_4 = \{1,3,5,6,7\}$ is not a 3-mps, since $M(A_2 \cup A_4) = 2 < 3$. Therefore, there are three 3-mps's {1,4,7}, {1,2,5,6,7} and {1,2,3,6,7}. For this particular flow network, its reliability for $k$=3 is discussed in details in Lee (1980) producing the same result.

## ACKNOWLEDGEMENT

We are grateful to the associate editor and the anonymous referee for valuable comments which improve the earlier version significantly.

## References

Aggarwal, K.K. (1988). "A Fast Algorithm for the Performance Index of a Telecommunication Network", *IEEE Transactions on Reliability*, **37**, 65-69.

Aggarwal, K.K., Chopra, Y.C. and Bajwa, J.S. (1982). "Capacity consideration in reliability analysis of communication systems", *IEEE Transactions on Reliability*, **31**, 177-181.

Barlow, R.E. and Proschan, F. (1982). *Statistical Theory of Reliability and Life Testing*, Holt, Rinehart & Winston.

Ford, L.R. and Fulkerson, D.R. (1962). *Flows in Networks*, Princeton University Press, Princeton, N.J.

Lee, S.H. (1980). "Reliability evaluation of a flow network", *IEEE Transactions on Reliability*, **29**, 24-26.

Locks, M.O. (1978). "Inverting and minimalizing path sets and cut sets", *IEEE Transactions on Reliability*, **27**, 107-109.

Misra, K.B. and Prasad, P. (1982). "Comments on: Reliability evaluation of a flow network", *IEEE Transactons on Reiability*, **31**, 174-176.

Qiu, L.Y. and Zhong, C.H. (1994). "A new algorithm for reliability evaluation of telecommunication networks with link-capacities", *Microelectronics & Reliability*, **34**, 1943-1946.

Rai, S. and Soh, S. (1991) "A component approach for reliability evaluation of telecommunication networks with heterogeneous link-capacities", *IEEE Transactions on Reliability*, **40**, 441-451.

Schanzer, R. (1995). "Comment on : Reliability modeling and performance of variable link-capacity networks", *IEEE Transactions on Reliability*, **44**, 620-621.

Varshney, P.K., Joshi, A.R. and Chang, P.L. (1994). "Reliability modeling and performance evaluation of variable link-capacity networks", *IEEE Transactions on Reliability*, **43**, 378-382.