International Journal of Innovative
Computing, Information and Control
Volume **8**, Number **6**, June **2012**

ICIC International ©2012 ISSN 1349-4198
pp. 4205–4217

EVALUATION OF NETWORK RELIABILITY FOR A COMPUTER NETWORK SUBJECT TO A BUDGET CONSTRAINT

Yi-Kuei Lin

Department of Industrial Management
National Taiwan University of Science and Technology
No. 43, Sec. 4, Keelung Rd., Taipei 106, Taiwan
yklin@mail.ntust.edu.tw

Received February 2011; revised June 2011

Abstract. *This paper constructs a multistate flow network composed of multistate edges to model a computer network. Each edge has lead time, capacity, and cost parameters. Therefore, the minimum transmission time through a single path is not a fixed number. Under the transmission protocol that the data are sent through p ($p \geq 2$) minimal paths simultaneously, the minimum transmission time is also stochastic. This paper is mainly to evaluate the probability that a given amount of data can be sent through p minimal paths simultaneously subject to both time and budget constraints. Such a probability is named network reliability herein, which can be treated as a performance indicator to measure the transmission capability of a computer network. Without knowing all minimal paths, a solution procedure is first proposed to calculate network reliability. Furthermore, the network administrator decides the routing rule indicating the first and the second priority p minimal paths in order to enhance the network reliability. Subsequently, network reliability according to the routing rule is also computed. At last, the expected demand, expected budget, expected time, and the criterion to find an ideal routing rule are presented as well.*

Keywords: Transmission protocol, Network reliability, Multistate flow network, Routing rule, Multiple minimal paths, Budget

1. **Introduction.** The shortest path problem to determine a path with minimum length is one of the well-known and practical problems in computer science, operations research, networking and other areas. This problem focuses on a network in which each edge has a fixed length parameter. When data/commodities are transmitted through a flow network, it is desirable to adopt the shortest path, least cost path, largest capacity path, shortest delay path, or some combination of multiple criteria [1-4], which are all variants of the shortest path problem. From the viewpoints of QoS (quality of service) [5-8] and business competing, it is an essential issue to shorten the transmission time through a computer network with the time parameter. Hence, a version of the shortest path problem called the quickest path problem proposed by Chen and Chin [9] arises to derive a single path with minimum transmission time for sending a given amount of data. Such a path is named the quickest path. In this problem, each edge has both capacity and lead time parameters [9-12]. The lead time is the time needed to travel through the edge. Several variants of the quickest path problems are thereafter proposed: constrained quickest path problem [13,14], the first k quickest paths problem [15-18], and all-pairs quickest path problem [19,20].

In the above problems, both the capacity and lead time parameters are assumed to be deterministic. However, due to failure, partial failure, maintenance, etc., each edge has multiple capacities/states in many real-life flow networks such as computer, pipelines transportation, logistics, telecommunication and urban traffic networks. Such a flow network is named a multistate flow network [21-29]. The minimum transmission time through a multistate flow network is thus not fixed. For instance, a pipelines transportation system with each edge representing the transmission medium is a typical multistate flow network. Virtually, each transmission medium consists of several pipelines, and each pipeline has only the normal/failure state. That implies a transmission medium has several states in which state k means that k pipelines are normal.

Besides, a computer network with each edge consisting of several cables such as coaxial cables and fiber optics, is also a multistate flow network. The transmission protocol, which allows the data to be transmitted through p ($p \geq 2$) disjoint minimal paths (MPs) simultaneously, shortens the transmission time. An MP, which is different from the so-called minimum path, is a path whose proper subsets are no longer paths. For convenience, we use p-MP to denote p disjoint MPs. For a multistate flow network, the transmission time problem to send d units of data (i.e., demand at the destination) through p-MP is never studied yet. Besides, cost is another crucial factor in business competing. The budget constraint is thus included in this paper. The network administrator needs some key performance indicators to measure QoS, especially the level to meet the customers' demand. The first addressed problem is to evaluate the probability that the multistate flow network can send d units of data through p-MP under both time and budget constraints. Such a probability named network reliability herein can be treated as a performance indicator to measure the transmission capability of a multistate flow network. For a specified p-MP, a simple algorithm is first proposed to generate all lower boundary vectors, the minimal capacity vectors sending d units of data under both time and budget constraints. Network reliability can then be calculated in terms of such vectors by applying inclusion-exclusion. In order to further enhance the transmission capability, the network administrator decides a routing rule in advance to indicate the first and the second priority p-MP. The second one will be responsible for the transmission duty if the first is out of order. The second addressed problem is to evaluate network reliability according to the routing rule. The remainder of this work is organized as follows. In Section 2, a multistate flow network with lead time, capacity, and cost parameters is established to model a computer network. The algorithm to generate all lower boundary vectors is subsequently proposed. The routing rule and network reliability are both presented in Section 3. An example is demonstrated in Section 4 to illustrate the algorithm and how network reliability may be calculated. Computational time complexity of the proposed algorithm is analyzed in Section 5. The expected demand, expected budget, expected time and more discussion about the routing rule are concluded in Section 6.

1.1. **Notations.**

p	number of MPs to send data simultaneously
T	time constraint
B	budget constraint
n	number of edges
a_i	edge $\#i$, $i = 1, 2, \ldots, n$
A	$\{a_i \| i = 1, 2, \ldots, n\}$: set of edges
N	set of nodes
l_i	lead time of a_i, $i = 1, 2, \ldots, n$
L	$(l_1, l_2, \ldots, l_n)$
M_i	maximal capacity of a_i, $i = 1, 2, \ldots, n$
M	$(M_1, M_2, \ldots, M_n)$
x_i	current capacity of a_i, $i = 1, 2, \ldots, n$

X	$(x_1, x_2, \ldots, x_n)$: capacity vector
r_i	number of possible capacities of a_i, $i = 1, 2, \ldots, n$
b_{ij}	jth possible capacity of a_i, $j = 1, 2, \ldots, r_i$. Thus, x_i takes possible values $0 = b_{i1} < b_{i2} < \ldots < b_{ir_i} = M_i$, $i = 1, 2, \ldots, n$
c_i	unit transmission cost on a_i per unit of flow, $i = 1, 2, \ldots, n$
C	$\{c_i \mid 1 \le i \le n\}$
G	(A, N, L, M, C): a multistate flow network
m	number of total MPs
P_s	MP $\#s$, $s = 1, 2, \ldots, m$
n_s	number of edges in P_s, $s = 1, 2, \ldots, m$
a_{se}	edge $\#e$ in P_s, $e = 1, 2, \ldots, n_s$
d	demand
$U(d_s, P_s)$	total cost for sending d units of data through P_s
d_s	assigned demand through P_s, $s = 1, 2, \ldots, m$
g	number of p-MP
Q_j	jth p-MP, $j = 1, 2, \ldots, g$
P_{jk}	MP $\#k$ in Q_j, $k = 1, 2, \ldots, p$. $Q_j = \{P_{j1}, P_{j2}, \ldots, P_{jp}\}$
$\mathbf{d}_j$	$(d_{j1}, d_{j2}, \ldots, d_{jp})$: assigned demand vector through Q_j
λ_j	set of feasible $\mathbf{d}_j$ subject to the budget constraint
$\lceil x \rceil$	smallest integer such that $\lceil x \rceil \ge x$
$\eta(d_s, X, P_s)$	transmission time to send d_s units of data through P_s under X and B
$\kappa(\mathbf{d}_j, X)$	minimum transmission time to send $\mathbf{d}_j$ under X
$\lambda(d, X, \mathrm{B}, Q_j)$	minimum transmission time to send d units of data through Q_j under X
$R_{d,\mathrm{T},\mathrm{B}}$	network reliability
$R_{d,\mathrm{T},\mathrm{B}}(\cdot)$	network reliability according to a routing rule
Φ_j	set of X sending d units of data through Q_j under T and B, $j = 1, 2, \ldots, g$
$\Phi_{j,\min}$	$\{X \mid X \text{ is minimal in } \Phi_j\}, j = 1, 2, \ldots, g$
h	number of lower boundary vectors for $(d, \mathrm{T}, \mathrm{B}, Q_j)$
S_k	subset of X, $k = 1, 2, \ldots, h$
$\overline{d_s}$	upper bound of d_s, $s = 1, 2, \ldots, m$
Γ_j	set of X generated from the algorithm
I, J	set of index
ρ	number of feasible $\mathbf{d}_j$
v_{jk}	necessary capacity for P_{jk} to send d_{jk} units of data under T, $k = 1, 2, \ldots, p$
V_t	tth $(v_{j1}, v_{j2}, \ldots, v_{jp})$
A_j	event that Q_j fails, $j = 1, 2, \ldots, g$
F_j	event that d units of data can be sent through Q_j under T and B, $j = 1, 2, \ldots, g$
E_i	a subset of X according to the cost rule, $i = 1, 2, \ldots, \mathrm{B}$
H_i	a subset of X according to the time rule, $i = 1, 2, \ldots, \mathrm{T}$

2. Problem Formulation and the Algorithm.

2.1. Assumptions.

(1) Each edge is multistate with a given probability distribution.
(2) Each node is perfectly reliable.

(3) The capacities of different edges are statistically independent.

2.2. **Constraints.** If d_s units of data are transmitted through an MP $P_s = \{a_{s1}, a_{s2}, \ldots, a_{sn_s}\}$, $s = 1, 2, \ldots, m$, then the total cost $U(d_s, P_s)$ is

$$U(d_s, P_s) = \sum_{e=1}^{n_s} (d_s \cdot c_{se}), \tag{1}$$

where $(d_s \cdot c_{se})$ is the cost through a_{se} for $e = 1, 2, \ldots, n_s$. The following equation states that the total cost for sending $\mathbf{d}_j$ through Q_j can not exceed the budget,

$$\sum_{k=1}^{p} U(d_{jk}, P_{jk}) \leq \mathrm{B}. \tag{2}$$

So, $\lambda_j = \{\mathbf{d}_j | \mathbf{d}_j$ satisfies constraints (2) and (3)$\}$ where

$$\sum_{k=1}^{p} d_{jk} = d. \tag{3}$$

The capacity of P_s under the capacity vector X is $\min\limits_{1 \leq e \leq n_s} (x_{se})$, $s = 1, 2, \ldots, m$. Hence, the transmission time to send d_s units of data through P_s under X, $\eta(d_s, X, P_s)$, is

$$\text{lead time of } P_s + \left\lceil \frac{d_s}{\text{capacity of } P_s} \right\rceil = \sum_{e=1}^{n_s} l_{se} + \left\lceil \frac{d_s}{\min\limits_{1 \leq e \leq n_s} x_{se}} \right\rceil. \tag{4}$$

The capacity vector X contradicts the time constraint if $\eta(d_s, X, P_s) > \mathrm{T}$. We have the result of Lemma 2.1.

Lemma 2.1. *$\eta(d_s, X, P_s) \geq \eta(d_s, Y, P_s)$ if $X < Y$ where Y is a capacity vector.*

Proof: If $X < Y$, then $x_{se} \leq y_{se}$ for each $a_{se} \in P_s$, and $\min\limits_{1 \leq e \leq n_s} x_{se} \leq \min\limits_{1 \leq e \leq n_s} y_{se}$. Thus, $\left\lceil \frac{d_s}{\min_{1 \leq e \leq n_s} x_{se}} \right\rceil \geq \left\lceil \frac{d_s}{\min_{1 \leq e \leq n_s} y_{se}} \right\rceil$. Since the lead time of P_s is independent of capacity vectors, we complete the proof by obtaining that $\eta(d_s, X, P_s) \geq \eta(d_s, Y, P_s)$. The proof is completed.

2.3. **Network reliability evaluation.** Any capacity vector X with $\lambda(d, X, \mathrm{B}, Q_j) \leq \mathrm{T}$ means that X can send d units of data through Q_j under both T and B. Thus, $\Phi_j = \{X | \lambda(d, X, \mathrm{B}, Q_j) \leq \mathrm{T}\}$. Network reliability $R_{d,\mathrm{T},\mathrm{B}}$ is the probability that G can send d units of data through a p-MP under both T and B. If the data are sent through Q_j, then

$$R_{d,\mathrm{T},\mathrm{B}} = \Pr\{X | \lambda(d, X, \mathrm{B}, Q_j) \leq \mathrm{T}\}, \tag{5}$$

where

$$\lambda(d, X, \mathrm{B}, Q_j) = \min_{\mathbf{d}_j \in \lambda_j} \kappa(\mathbf{d}_j, X), \tag{6}$$

and

$$\kappa(\mathbf{d}_j, X) = \max_{1 \leq k \leq p} \eta(d_{jk}, X, p_{jk}). \tag{7}$$

If the network size is large, the number of $X \in \Phi_j$ will be enormous. We thus propose an idea about lower boundary vectors and subsets as follows.

Definition 2.1. *$X \in \Phi_{j,\min}$ is called a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. Or equivalently, X is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$ if and only if (i) $\lambda(d, X, \mathrm{B}, Q_j) \leq \mathrm{T}$ and (ii) $\lambda(d, Y, \mathrm{B}, Q_j) > \mathrm{T}$ for any capacity vector Y with $Y < X$.*

The following lemma further shows that any capacity vector, which is larger than a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$, satisfies both time and budget constraints.

Lemma 2.2. *If X is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$, then $Y \in \Phi_j$ for any $Y > X$.*

Proof: Since X is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$, there exists a $\mathbf{d}_j \in \lambda_j$ such that $\kappa(\mathbf{d}_j, X) = \lambda(d, X, \mathrm{B}, Q_j) \leq \mathrm{T}$ and $\sum_{k=1}^{p} U(d_{jk}, P_{jk}) \leq \mathrm{B}$. Lemma 2.1 states that $\eta(d_s, Y, P_s) \leq \eta(d_s, X, P_s)$ for any $Y > X$. Hence, $\max_{1 \leq k \leq p} \eta(d_{jk}, Y, p_{jk}) \leq \max_{1 \leq k \leq p} \eta(d_{jk}, X, p_{jk})$, equivalently, $\kappa(\mathbf{d}_j, Y) \leq \kappa(\mathbf{d}_j, X)$. Then $\min_{\mathbf{d}_j \in \lambda_j} \kappa(\mathbf{d}_j, Y) \leq \min_{\mathbf{d}_j \in \lambda_j} \kappa(\mathbf{d}_j, X)$. We conclude that $Y \in \Phi_j$ by obtaining $\lambda(d, Y, \mathrm{B}, Q_j) \leq \lambda(d, X, \mathrm{B}, Q_j) \leq \mathrm{T}$. The proof is completed.

Suppose $X_1, X_2, \ldots, X_h$ are all lower boundary vectors for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. Lemma 2.2 implies that $\Pr\{X | \lambda(d, X, \mathrm{B}, Q_j) \leq \mathrm{T}\} = \Pr\{X | X \geq X_k$ for a lower boundary vector X_k for $(d, \mathrm{T}, \mathrm{B}, Q_j)\}$. Hence, $R_{d,\mathrm{T},\mathrm{B}}$ could be represented as the union of subsets,

$$\begin{aligned} R_{d,\mathrm{T},\mathrm{B}} &= \Pr\left\{ \bigcup_{k=1}^{h} \{X | X \geq X_k\} \right\} \\ &= \Pr\left\{ \bigcup_{k=1}^{h} S_k \right\}. \end{aligned} \tag{8}$$

Several methods such as inclusion-exclusion [23-27,30,31], disjoint-event method [30,32], and state-space decomposition [21,22,33,34] may be applied to calculate $\Pr\left\{ \bigcup_{k=1}^{h} S_k \right\}$.

2.4. **Algorithm.**

Algorithm 1. Generate all lower boundary vectors for $(d, \mathrm{T}, \mathrm{B}, Q_j)$.

1 for $P_{jk} \in Q_j$, $k = 1, 2, \ldots, p$, find the largest number $\overline{d_{jk}}$ subject to

$$\sum_{i:a_i \in P_{jk}} l_i + \left\lceil \frac{\overline{d_{jk}}}{\min_{i:a_i \in P_{jk}} M_i} \right\rceil \leq \mathrm{T} \tag{9}$$

2 find all integer solutions $\mathbf{d}_j = (d_{j1}, d_{j2}, \ldots, d_{jk})$ of

$$\sum_{k=1}^{p} d_{jk} = d \text{ subject to } d_{jk} \leq \overline{d_{jk}}, \quad k = 1, 2, \ldots, p \tag{10}$$

//generate λ_j

3 $\Gamma_j = \varnothing$, $I = \varnothing$, $J = \varnothing$. Suppose there are ρ feasible $\mathbf{d}_j$.

4 **for** $t = 1$ to ρ, **do**

//do the following steps for tth $\mathbf{d}_j$

5 calculate $\sum_{k=1}^{p} U(d_{jk}, P_{jk})$

6 **if** $\sum_{k=1}^{p} U(d_{jk}, P_{jk}) > \mathrm{B}$, **then** goto next t

7 **for** $k = 1$ to p, **do**

8 find the smallest integer v_{jk} such that

$$\sum_{i:a_i \in P_{jk}} l_i + \left\lceil \frac{d_{jk}}{v_{jk}} \right\rceil \leq \mathrm{T} \tag{11}$$

9 **end**
10 $V_t = (v_{j1}, v_{j2}, \ldots, v_{jp})$
11 **for** $w = 1$ to $t-1$ and $w \notin \mathrm{I} \cup \mathrm{J}$, **do**
12 **if** $V_t \geq V_w$, **then** $\mathrm{I} = \mathrm{I} \cup \{\mathrm{t}\}$ and goto next t
13 **if** $V_t < V_w$, **then** $\mathrm{I} = \mathrm{I} \cup \{\mathrm{w}\}$ and $\Gamma_j = \Gamma_j \backslash \{X_w\}$
14 **end**
15 $X_t = (x_1, x_2, \ldots, x_n)$ where

$$x_i = \begin{cases} \text{minimal capacity } u \text{ of } a_i \text{ such that } u \geq v_{jk} & \text{if } a_i \in P_{jk},\ k = 1, 2, \ldots, p \\ 0 & \text{others} \end{cases} \tag{12}$$

16 **for** $w = 1$ to $t-1$ and $w \notin \mathrm{I} \cup \mathrm{J}$, **do**
17 **if** $X_t \geq X_w$, **then** $\mathrm{J} = \mathrm{J} \cup \{t\}$ and goto next t;
18 **if** $X_t < X_w$, **then** $\mathrm{J} = \mathrm{J} \cup \{w\}$ and $\Gamma_j = \Gamma_j \backslash \{X_w\}$
19 **end**
20 $\Gamma_j = \Gamma_j \cup \{X_t\}$
21 **end**

Steps 1 and 2 generate the set λ_j, and Steps 5 to 15 guarantee that $\lambda(d, X_t, \mathrm{B}, Q_j) \leq \mathrm{T}$ because $\eta(d_{jk}, X_t, P_{jk}) \leq \mathrm{T}$ for each $k = 1, 2, \ldots, p$ (Step 8). The X_t (in Step 15) is thus a candidate of lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. Steps 16 to 19 further check the qualification of the candidates, and the set J stores the index of non-minimal X_t. To make it clear that $\Gamma_j = \Phi_{j,\min}$, the following theories are essential.

Lemma 2.3. *If $X_t \in \Gamma_j$, then $X_t \in \Phi_j$.*

Proof: It is known that $\eta(d_{jk}, X_t, P_{jk}) = \sum_{i:a_i \in P_{jk}} l_i + \left\lceil \frac{d_{jk}}{x_{jk}} \right\rceil \leq \sum_{i:a_i \in P_{jk}} l_i + \left\lceil \frac{d_{jk}}{v_{jk}} \right\rceil \leq \mathrm{T}$ for each k since $x_{jk} \geq v_{jk}$. So $\kappa(\mathbf{d}_j, X_t) = \max_{1 \leq k \leq p} \eta(d_{jk}, X_t, p_{jk}) \leq \mathrm{T}$. The proof is completed by obtaining $\lambda(d, X_t, \mathrm{B}, Q_j) = \min_{\mathbf{d}_j \in \lambda_j} \kappa(\mathbf{d}_j, X_t) \leq \mathrm{T}$. The proof is completed.

Theorem 2.1. *Each $X_t \in \Gamma_j$ is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$.*

Proof: By Lemma 2.3, it is known that $X_t \in \Phi_j$. Suppose $X_t \notin \Phi_{j,\min}$. Then there exists a lower boundary vector $Y = (y_1, y_2, \ldots, y_n)$ for $(d, \mathrm{T}, \mathrm{B}, Q_j)$ such that $Y < X_t$. Without loss of generality, there exists an edge $a_u \in P_{j1}$ such that $y_u < x_u$. It is known that x_u is the minimal capacity of a_u such that $x_u \geq v_{j1}$. The situation $y_u < x_u$ results in that $y_u < v_{j1}$ and $\sum_{i:a_i \in P_{j1}} l_i + \left\lceil \frac{d_{j1}}{y_u} \right\rceil > \mathrm{T}$. It contradicts that Y is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. Hence, X_t is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. The proof is completed.

Theorem 2.2. *Each lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$ belongs to Γ_j.*

Proof: Let X be a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. We claim that X satisfies Equation (12). Firstly, if there exists an edge $a_e \in \bigcup_{k=1}^{p} P_{jk}$ such that $x_e > f$, then set $Z = (x_1, x_2, \ldots, x_e - f, \ldots, x_n)$, where $(x_e - f)$ is the maximal capacity of a_e such that $(x_e - f) < x_e$. Thus, $\eta(d_{jk}, Z, P_{jk}) \leq \eta(d_{jk}, X, P_{jk}) \leq \mathrm{T}$, $k = 1, 2, \ldots, p$. That is, $Z \in \Phi_j$ which contradicts that X is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. Secondly, if there exists an edge $a_w \notin \bigcup_{k=1}^{p} P_{jk}$ such that $x_w > 0$, then set $Y = (x_1, x_2, \ldots, x_w - z, \ldots, x_n)$, where $(x_w - z)$ is the maximal capacity of a_w such that $(x_w - z) < x_w$. Thus, $\eta(d_{jk}, Y, P_{jk}) = \eta(d_{jk}, X, P_{jk}) \le \mathrm{T}$, $k = 1, 2, \ldots, p$, since $a_w \notin \bigcup_{k=1}^{p} P_{jk}$. Then $Y \in \Phi_j$ which contradicts that X is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. Hence, $x_i = 0$ for each $a_i \notin \bigcup_{k=1}^{p} P_{jk}$. The above proves that X satisfies Equation (12).

We further claim that $X \in \Gamma_j$. Suppose to the contrary that $X \notin \Gamma_j$. Then there exists a $W \in \Gamma_j$ such that $W < X$ because Γ_j is generated from the X satisfying Equation (12). Theorem 2.1 indicates that W is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. It contradicts the fact that X is a lower boundary vector for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. Therefore, we conclude that $X \in \Gamma_j$. The proof is completed.

3. **Network Reliability According to the Routing Rule.** The routing rule, a transmission rule decided by the network administrator, indicates the first priority p-MP, the second priority (or named alternative) p-MP, etc. The routing level is called 2 (resp. 3) if an (resp. two) alternative p-MP is standing by. The second priority p-MP takes charge of the transmission duty if the first fails; and the third takes charge if the second fails.

Definition 3.1. *An MP fails if and only if at least one edge in it fails.*

Definition 3.2. *A p-MP fails if and only if all MPs in it fail.*

Then, the probability that Q_j fails is

$$\begin{aligned}\Pr(A_j) &= \Pr(P_{jk} \text{ fail for each } k = 1, 2, \ldots, p)\\ &= \Pr(x_i = 0 \text{ for at least one } a_i \in P_{j1}) \times \Pr(x_i = 0 \text{ for at least one}\\ &\quad\ a_i \in P_{j2}) \times \ldots \times \Pr(x_i = 0 \text{ for at least one } a_i \in P_{jp})\\ &= \prod_{k=1}^{p}\left(1 - \prod_{i:a_i \in P_{jk}} \Pr(x_i \ge 1)\right), \quad j = 1, 2, \ldots, g. \qquad (13)\end{aligned}$$

Theorems 2.1 and 2.2 prove that $\Phi_{j,\min}$ can be generated from the algorithm. Thus,

$$\Pr(F_j) = \Pr\{\Phi_{j,\min}\} = \Pr\left\{\bigcup_{k=1}^{h} S_k\right\}, \quad j = 1, 2, \ldots, g. \qquad (14)$$

$\Pr(F_j)$ is also network reliability if the data are sent only through Q_j. We first concentrate on routing level 2. Without loss of generality, let Q_1 and Q_2 be the first and the second priority p-MP, respectively. Network reliability with routing level 2 is thus

$$\begin{aligned}&R_{d,\mathrm{T},\mathrm{B}}(Q_1, Q_2)\\ &= \Pr(F_1) + \Pr(F_2|A_1) \times \Pr(A_1)\\ &= \Pr(F_1) + \Pr(F_2) \times \Pr(A_1), \qquad (15)\end{aligned}$$

where $\Pr(F_2|A_1) = \Pr(F_2)$ since MPs and Q_j are both statistically independent from Assumption (3).

4. **Numerical Example.** We use a computer network shown in Figure 1 to illustrate the proposed solution procedure for $p = 2$. The edge data are all shown in Table 1. Let Q_1 consist of $P_1 = \{a_1, a_2, a_3\}$ and $P_2 = \{a_4, a_5, a_6\}$.

TABLE 1. The edge data of Figure 1

edge	capacity	probability	lead time	cost	edge	capacity	probability	lead time	cost
a_1	50[a]	0.86	2	3	a_{12}	60	0.83	2	2
	30	0.05				40	0.04		
	10	0.03				20	0.04		
	0	0.06				10	0.05		
a_2	50	0.90	2	4		0	0.04		
	30	0.03			a_{13}	60	0.82	1	3
	10	0.02				40	0.10		
	0	0.05				20	0.02		
a_3	40	0.88	3	3		10	0.02		
	20	0.04				0	0.04		
	10	0.04			a_{14}	20	0.95	2	2
	0	0.04				0	0.05		
a_4	50	0.85	3	2	a_{15}	70	0.80	3	2
	30	0.05				50	0.05		
	10	0.05				30	0.05		
	0	0.05				10	0.05		
a_5	50	0.85	4	3		0	0.05		
	30	0.05			a_{16}	60	0.82	2	2
	10	0.05				40	0.06		
	0	0.05				30	0.04		
a_6	40	0.88	3	2		10	0.04		
	20	0.04				0	0.04		
	10	0.03			a_{17}	50	0.88	3	3
	0	0.05				30	0.03		
a_7	20	0.96	2	1		10	0.04		
	0	0.04				0	0.05		
a_8	50	0.85	3	3	a_{18}	40	0.85	3	4
	30	0.05				30	0.05		
	10	0.05				10	0.05		
	0	0.05				0	0.05		
a_9	40	0.83	4	1	a_{19}	50	0.90	3	1
	20	0.07				30	0.03		
	10	0.05				10	0.03		
	0	0.05				0	0.04		
a_{10}	40	0.85	2	2	a_{20}	40	0.80	3	2
	20	0.05				20	0.10		
	10	0.05				10	0.05		
	0	0.05				0	0.05		
a_{11}	50	0.88	3	1	a_{21}	20	0.93	2	2
	30	0.02				0	0.07		
	10	0.05			a_{22}	10	0.96	4	1
	0	0.05				0	0.04		

[a] $\Pr\{\text{the capacity of } a_1 \text{ is } 50\} = 0.86$.

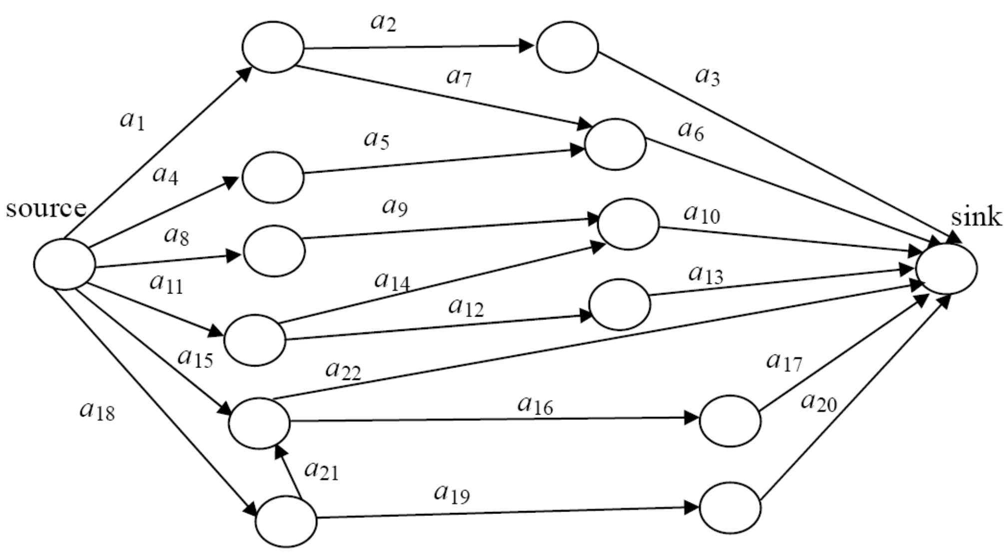

FIGURE 1. A computer network

4.1. **Case I.** We would like to evaluate network reliability that 200 units of data can be sent through Q_1 under both time 13 and budget 2000. All lower boundary vectors for $(200, 13, 2000, Q_1)$ can be derived as the following steps.

1 The largest demand $\overline{d_1}$ such that $(2+2+3) + \left\lceil \frac{\overline{d_1}}{\min\{50,50,40\}} \right\rceil \le 13$ is 200. Similarly, the largest demand $\overline{d_2}$ is 120.

2 Totally, 13 feasible $\mathbf{d}_1 = (d_1, d_2)$ are obtained from of $d_1 + d_2 = 200$ subject to $d_1 \le 200$ and $d_2 \le 120$, supposing that data are sent in packages of 10 units. The results are shown in Table 2.

3 $\Gamma_j = \varnothing$, $\mathrm{I} = \varnothing$, $\mathrm{J} = \varnothing$, $\rho = 13$.

4 $t = 1$, $\mathbf{d}_1 = (80, 120)$.

5 $U(80, P_1) = 80 \times (3+4+3) = 800$ and $U(120, P_2) = 840$.

6 $U(80, P_1) + U(120, P_2) < 2000$.

7 For $k = 1$ and 2

8 Then $v_1 = 20$ is the smallest integer such that $\left(7 + \left\lceil \frac{80}{v_1} \right\rceil\right) \le 13$. Similarly, $v_2 = 40$.

10 $V_1 = (v_1, v_2) = (20, 40)$, $\mathrm{I} = \varnothing$.

15 $X_1 = (30, 30, 20, 50, 50, 40, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$, $\mathrm{J} = \varnothing$.

20 $\Gamma_j = \Gamma_j \cup \{X_1\} = \{X_1\}$.

4.1 $t = 2$, $\mathbf{d}_1 = (90, 110)$.

5.1 $U(90, P_1) = 90 \times (3+4+3) = 900$ and $U(110, P_2) = 110 \times (2+3+2) = 770$.

6.1 $U(90, P_1) + U(110, P_2) < 2000$.

7.1 For $k = 1$ and 2.

8.1 $v_1 = 20$ is the smallest integer such that $\left(7 + \left\lceil \frac{90}{v_1} \right\rceil\right) \le 13$. Similarly, $v_2 = 40$.

10.1 $V_2 = (20, 40)$.

12.1 $\mathrm{I} = \mathrm{I} \cup \{2\} = \{2\}$ since $V_2 \ge V_1$.

4.2 $t = 3$, $\mathbf{d}_1 = (100, 100)$.

$\vdots$

The results are summarized in Table 2.

Table 2. Results of running the algorithm for Q_1

(d_1, d_2)	$U(d_1, P_1)$	$U(d_2, P_2)$	Total cost	$V_t = (v_1, v_2)$	X	$X \in \Gamma_1$?	Remark
$(80, 120)$	800	840	1640	$V_1 = (2, 4)$			$V_1 > V_4$
(90,110)	900	770	1670	$V_2 = (2, 4)$			$V_2 \geq V_1$
$(100, 100)$	1000	700	1700	$V_3 = (2, 4)$			$V_3 \geq V_1$
$(110, 90)$	1100	630	1730	$V_4 = (2, 3)$	$X_4 = (3, 3, 2, 3, 3, 4, 0, 0, \ldots, 0)$	YES	
$(120, 80)$	1200	560	1760	$V_5 = (2, 3)$			$V_5 \geq V_4$
$(130, 70)$	1300	490	1790	$V_6 = (3, 3)$			$V_6 \geq V_4$
$(140, 60)$	1400	420	1820	$V_7 = (3, 2)$			$V_7 > V_{10}$
$(150, 50)$	1500	350	1850	$V_8 = (3, 2)$			$V_8 \geq V_7$
$(160, 40)$	1600	280	1880	$V_9 = (3, 2)$			$V_9 \geq V_7$
$(170, 30)$	1700	210	1910	$V_{10} = (3, 1)$	$X_{10} = (3, 3, 4, 1, 1, 1, 0, 0, \ldots, 0)$	YES	
$(180, 20)$	1800	140	1940	$V_{11} = (3, 1)$			$V_{11} \geq V_{10}$
$(190, 10)$	1900	70	1970	$V_{12} = (4, 1)$			$V_{12} > V_{10}$
$(200, 0)$	2000	0	2000	$V_{13} = (4, 0)$	$X_{13} = (5, 5, 4, 0, 0, 0, 0, 0, \ldots, 0)$	YES	

Three lower boundary vectors for $(200, 13, 2000, Q_1)$ are generated: $X_4 = (3, 3, 2, 3, 3, 4, 0, 0, \ldots, 0)$, $X_{10} = (3, 3, 4, 1, 1, 1, 0, 0, \ldots, 0)$, and $X_{13} = (5, 5, 4, 0, 0, 0, 0, 0, \ldots, 0)$. Let $S_1 = \{X|X \geq X_4\}$, $S_2 = \{X|X \geq X_{10}\}$, and $S_3 = \{X|X \geq X_{13}\}$. Then network reliability $R_{200,13,2000} = \Pr\{S_1 \cup S_2 \cup S_3\}$ is 0.759799331.

4.2. **Case II.** For another case that the data are sent through another $Q_2 = \{P_3, P_4\}$ where $P_3 = \{a_8, a_9, a_{10}\}$ and $P_4 = \{a_{11}, a_{12}, a_{13}\}$, the results described in Table 3 shows that three lower boundary vectors for $(200, 13, 2000, Q_2)$ are generated: X_1, X_7 and X_{14}. Network reliability $R_{200,13,2000}$ increases to be 0.81980716.

Table 3. Results of running the algorithm for Q_2

(d_3, d_4)	$U(d_3, P_3)$	$U(d_4, P_4)$	Total cost	$V_t = (v_3, v_4)$	X	$X \in \Gamma_2$?	Remark
$(0, 200)$	0	1200	1200	$V_1 = (0, 3)$	$X_1 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 4, 4, 0, \ldots, 0)$	YES	
$(10, 190)$	60	1140	1200	$V_2 = (1, 3)$			$V_2 > V_1$
$(20, 180)$	120	1080	1200	$V_3 = (1, 3)$			$V_3 > V_1$
$(30, 170)$	180	1020	1200	$V_4 = (1, 3)$			$V_4 > V_1$
$(40, 160)$	240	960	1200	$V_5 = (1, 3)$			$V_5 > V_1$
$(50, 150)$	300	900	1200	$V_6 = (2, 3)$			$V_6 > V_1$
$(60, 140)$	360	840	1200	$V_7 = (2, 2)$	$X_7 = (0, 0, 0, 0, 0, 0, 0, 3, 2, 2, 3, 2, 2, 0, \ldots, 0)$	YES	
$(70, 130)$	420	780	1200	$V_8 = (2, 2)$			$V_8 \geq V_7$
$(80, 120)$	480	720	1200	$V_9 = (2, 2)$			$V_9 \geq V_7$
$(90, 110)$	540	660	1200	$V_{10} = (3, 2)$			$V_{10} > V_7$
$(100, 100)$	600	600	1200	$V_{11} = (3, 2)$			$V_{11} > V_7$
$(110, 90)$	660	540	1200	$V_{12} = (3, 2)$			$V_{12} > V_7$
$(120, 80)$	720	480	1200	$V_{13} = (4, 2)$			$V_{13} > V_7$
$(130, 70)$	780	420	1200	$V_{14} = (4, 1)$	$X_{14} = (0, 0, 0, 0, 0, 0, 0, 5, 4, 4, 1, 1, 1, 0, \ldots, 0)$	YES	
$(140, 60)$	840	360	1200	$V_{15} = (4, 1)$			$V_{15} \geq V_{14}$
$(150, 50)$	900	300	1200	$V_{16} = (4, 1)$			$V_{16} \geq V_{14}$
$(160, 40)$	960	240	1200	$V_{17} = (4, 1)$			$V_{17} \geq V_{14}$

4.3. **Network reliability with routing level 2.** Furthermore, according to the routing rule that Q_1 and Q_2 are the first and the second priority p-MP, respectively, network reliability is

$$\begin{aligned} R_{200,13,2000}(Q_1, Q_2) &= \Pr(F_1) + \Pr(F_2) \times \Pr(A_1) \\ &= 0.759799331 + 0.81980716 \times 0.02035544 = 0.776486866. \end{aligned}$$

Comparing with the case that the data are sent only through Q_1, the routing rule with Q_1 and Q_2 shows an increase of 0.016687535 on reliability.

4.4. **Network reliability with routing level 3.** We further extend the solution procedure to routing level 3. If Q_1, Q_2 and Q_3 are the first, the second, and the third priority p-MP, respectively, then network reliability is

$$\begin{aligned} & R_{d,\mathrm{T},\mathrm{B}}(Q_1, Q_2, Q_3) \\ &= \Pr(F_1) + \Pr(F_2|A_1) \times \Pr(A_1) + \Pr(F_3|A_1A_2) \times \Pr(A_1A_2) \\ &= \Pr(F_1) + \Pr(F_2) \times \Pr(A_1) + \Pr(F_3) \times \Pr(A_1) \times \Pr(A_2). \end{aligned} \tag{16}$$

By utilizing the example data, we obtain

$$\begin{aligned} R_{200,13,2000}(Q_1, Q_2, Q_3) = & 0.759799331 + 0.81980716 \times 0.02035544 \\ & + 0.791119744 \times 0.02035544 \times 0.01775396 = 0.776772769, \end{aligned}$$

which is greater than $R_{200,13,2000}(Q_1, Q_2)$.

5. **Computational Time Complexity.** The algorithm takes at most $O(n)$ time to find the largest assigned demand $\overline{d_{jk}}$ for all $P_{jk} \in Q_j$ because each p-MP has no more than n edges. For each $\mathbf{d}_j$, it spends $O(n)$ time to test the budget and time constraints in the worst case, respectively. Hence, we need at most $O(\rho n)$ time for all $\mathbf{d}_j$. It subsequently needs at most $O(\rho)$ time to compare with other V_w for each V_t and at most $O(\rho^2)$ time for all V_t. The transformation from V_t to X_t takes $O(n)$ time in the worst case. Each X_t needs $O(\rho n)$ time to compare with other X_w in the worst case because Γ_j contains at most ρ elements. Hence, it needs $O(\rho^2 n)$ time in the worst case to generate all lower boundary vectors for $(d, \mathrm{T}, \mathrm{B}, Q_j)$. In sum, it takes at most $O(\rho^2 n)$ time to execute the proposed algorithm. It seems that $O(10000n)$ time is needed if $d = 100$. In practice, the data amount (d_{j1}, d_{j2}) may be $(100, 0), (90, 10), \ldots, (0, 100)$ if $p = 2$. The number of feasible $\mathbf{d}_j$ is 11 but not 101. Thus only $O(121n)$ time is needed to execute the algorithm.

6. **Discussion and Conclusion.** According to $R_{d,\mathrm{T},\mathrm{B}} = \Pr\{X|X \in \Phi_j\}$, the corresponding total cost for each X does not exceed B. We may divide Φ_j into disjoint subset E_i, $i = 1, 2, \ldots, \mathrm{B}$, where E_i is the set of X whose total cost equals i. Although different $\mathbf{d}_j$ may generate the same X, we can set the total cost of X as the minimal cost $\sum_{k=1}^{p} U(d_{jk}, P_{jk})$ from those corresponding $\mathbf{d}_j$. Hence,

$$R_{d,\mathrm{T},\mathrm{B}} = \sum_{i=1}^{\mathrm{B}} \Pr\{E_i\}, \tag{17}$$

and

$$R_{d,\mathrm{T},\mathrm{B}} - R_{d,\mathrm{T},\mathrm{B}-1} = \Pr\{E_{\mathrm{B}}\}. \tag{18}$$

Similarly, Φ_j is divided into disjoint subset H_i, $i = 1, 2, \ldots, \mathrm{T}$, where H_i is the set of X whose total time equals i. Hence,

$$R_{d,\mathrm{T},\mathrm{B}} = \sum_{i=1}^{\mathrm{T}} \Pr\{H_i\}, \tag{19}$$

and

$$R_{d,\mathrm{T},\mathrm{B}} - R_{d,\mathrm{T}-1,\mathrm{B}} = \Pr\{H_{\mathrm{T}}\}. \tag{20}$$

Under the same condition that (time, budget) $= (13, 2000)$, Table 4 lists network reliabilities with respect to all combination of priorities. It is known that $\mathrm{P}(F_1) = 0.759799331$, $\mathrm{P}(F_2) = 0.81980716$ and $\mathrm{P}(F_3) = 0.791119744$. According to the sorting criterion, the routing rule that Q_2 and Q_3 are the first and the second priority p-MP, respectively, reflects the highest network reliability 0.833852668.

Table 4. Network reliabilities with respect to different priorities

1st priority p-MP	Q_1	Q_2	Q_1	Q_3	Q_2	Q_3
2nd priority p-MP	Q_2	Q_1	Q_3	Q_1	Q_3	Q_2
network reliability	0.776486866	0.833296606	0.775902921	0.804681371	0.833852668	0.805752449

From the point of view of quality management, we obtain the following practical usage of the theoretic results.

1. Network reliability is a performance indicator to measure the transmission capability. Such an indicator reflects the level of QoS for a computer network.
2. The increase (or contribution) on network reliability by the backup p-MP can be computed easily. The supervisor can judge the contribution of one or more backup p-MPs.
3. Sensitivity analysis can be executed to improve the most important component (e.g., switch or server in computer network) which increases/decreases network reliability most significantly.

In this paper, we assume each Q_j consists of p disjoint MPs and $Q_i \cap Q_j = \varnothing$ if $i \neq j$. The further research can study the routing rule that Q_j consists of at least two intersectional MPs, and the case that some MPs are included in different Q_j. Moreover, it would be worthwhile to find the optimal routing rule with highest network reliability.

Acknowledgements. This work was supported in part by the National Science Council, Taiwan, under Grant No. NSC 98-2221-E-011-051-MY3.

REFERENCES

[1] R. K. Ahuja, Minimum cost-reliability ratio problem, *Computers and Operations Research*, vol.16, pp.83-89, 1998.
[2] L. D. Bodin, B. L. Golden, A. A. Assad and M. O. Ball, Routing and scheduling of vehicles and crews: The state of the art, *Computers and Operations Research*, vol.10, pp.63-211, 1982.
[3] M. L. Fredman and R. E. Tarjan, Fibonacci heaps and their uses in improved network optimization algorithms, *J. ACM*, vol.34, pp.596-615, 1987.
[4] B. L. Golden and T. L. Magnanti, Deterministic network optimization: A bibliography, *Networks*, vol.7, pp.149-183, 1977.
[5] R.-C. Chen and C.-M. Lin, Adaptive end-to-end delay equalizations for tcp virtual path transmissions in Internet environments, *International Journal of Innovative Computing, Information and Control*, vol.6, no.3(A), pp.1069-1078, 2010.
[6] G. Lee, Z.-K. Lee, H. Song and S. H. Park, Contention-aware qos routing algorithm based on multi-rate service of IEEE 802.11 over mobile ad hoc networks, *International Journal of Innovative Computing, Information and Control*, vol.6, no.3(B), pp.1221-1229, 2010.
[7] V. S. Tseng, M.-H. Hsieh and K. W. Lin, A novel cache replacement algorithm for cooperative caching in wireless multimedia sensor networks, *International Journal of Innovative Computing, Information and Control*, vol.7, no.2, pp.763-776, 2011.
[8] J.-F. Lin, Scheduling parallel tasks with intra-communication overhead in a grid computing environment, *International Journal of Innovative Computing, Information and Control*, vol.7, no.2, pp.881-896, 2011.
[9] Y. L. Chen and Y. H. Chin, The quickest path problem, *Computers and Operations Research*, vol.17, pp.153-161, 1990.
[10] Y. C. Hung and G. H. Chen, Distributed algorithms for the quickest path problem, *Parallel Computing*, vol.18, pp.823-834, 1992.
[11] E. D. Q. V. Martins and J. L. E. D. Santos, An algorithm for the quickest path problem, *Operations Research Letters*, vol.20, pp.195-198, 1997.

[12] C. K. Park, S. Lee and S. Park, A label-setting algorithm for finding a quickest path, *Computers and Operations Research*, vol.31, pp.2405-2418, 2004.
[13] G. H. Chen and Y. C. Hung, Algorithms for the constrained quickest path problem and the enumeration of quickest paths, *Computers and Operations Research*, vol.21, pp.113-118, 1994.
[14] Y. L. Chen and K. Tang, Minimum time paths in a network with mixed time constraints, *Computers and Operations Research*, vol.25, pp.793-805, 1998.
[15] Y. L. Chen, An algorithm for finding the k quickest paths in a network, *Computers and Operations Research*, vol.20, pp.59-65, 1993.
[16] Y. L. Chen, Finding the k quickest simples paths in a network, *Information Processing Letters*, vol.50, pp.89-92, 1994.
[17] J. C. N. Clímaco, M. M. B. Pascoal, J. M. F. Craveirinha and M. E. V. Captivo, Internet packet routing: Application of a K-quickest path algorithm, *European Journal of Operational Research*, vol.181, pp.1045-1054, 2007.
[18] M. M. B. Pascoal, M. E. V. Captivo and J. C. N. Clímaco, An algorithm for ranking quickest simple paths, *Computers and Operations Research*, vol.32, pp.509-520, 2005.
[19] G. H. Chen and Y. C. Hung, On the quickest path problem, *Information Processing Letters*, vol.46, pp.125-128, 1993.
[20] D. T. Lee and E. Papadopoulou, The all-pairs quickest path problem, *Information Processing Letters*, vol.45, pp.261-267, 1993.
[21] C. C. Jane, J. S. Lin and J. Yuan, On reliability evaluation of a limited-flow network in terms of minimal cutsets, *IEEE Transactions on Reliability*, vol.42, pp.354-361, 1993.
[22] J. S. Lin, C. C. Jane and J. Yuan, On reliability evaluation of a capacitated-flow network in terms of minimal pathsets, *Network*, vol.25, pp.131-138, 1995.
[23] Y. K. Lin, Extend the quickest path problem to the system reliability evaluation for a stochastic-flow network, *Computers and Operations Research*, vol.30, pp.567-575, 2003.
[24] Y. K. Lin, Reliability evaluation for an information network with node failure under cost constraint, *IEEE Transactions on Systems, Man and Cybernetics – Part A: Systems and Humans*, vol.37, pp.180-188, 2007.
[25] Y. K. Lin, On a multicommodity stochastic-flow network with unreliable nodes subject to budget constraint, *European Journal of Operational Research*, vol.176, pp.347-360, 2007.
[26] Y. K. Lin and C. T. Yeh, Evaluation of optimal network reliability under components assignments subject to a transmission budget, *IEEE Transactions on Reliability*, vol.59, pp.539-550, 2010.
[27] Y. K. Lin and C. T. Yeh, Optimal resources assignment to maximize multistate network reliability, *Computers and Operations Research*, vol.37, pp.2229-2238, 2010.
[28] W. C. Yeh, An improved sum-of-disjoint-products technique for the symbolic network reliability analysis with known minimal paths, *Reliability Engineering and System Safety*, vol.92, pp.260-268, 2007.
[29] W. C. Yeh, A simple minimal path method for estimating the weighted multi-commodity multistate unreliable networks reliability, *Reliability Engineering and System Safety*, vol.93, pp.125-136, 2008.
[30] J. C. Hudson and K. C. Kapur, Reliability bounds for multistate systems with multistate components, *Operations Research*, vol.33, pp.153-160, 1985.
[31] J. Xue, On multistate system analysis, *IEEE Transactions on Reliability*, vol.34, pp.329-337, 1985.
[32] R. Yarlagadda and J. Hershey, Fast algorithm for computing the reliability of communication network, *Int. J. Electronic*, vol.70, pp.549-564, 1991.
[33] C. Alexopoulos, A note on state-space decomposition methods for analyzing stochastic flow networks, *IEEE Transactions on Reliability*, vol.44, pp.354-357, 1995.
[34] T. Aven, Reliability evaluation of multistate systems with multistate components, *IEEE Transactions on Reliability*, vol.34, pp.473-479, 1985.