

네트워크 신뢰성 중요도의 척도와 계산알고리즘[†]

(A Measure of Reliability Importance in the Network and its Computational Algorithm)

이 창 훈 *
정 성 진 *
홍 정 식 **

Abstract

A unified reliability importance of the component in the network is considered. A measure is presented by considering the traffic between vertex pairs. An algorithm for this measure is also presented by using the factoring algorithm.

I. 서 론

컴퓨터와 수송수단의 발달로 인하여 네트워크 시스템이 계속 증대되고 있다.^[1] 이러한 네트워크 시스템은 일반 시스템과 비교할 때 몇 가지 특성을 가지고 있다. 첫째는 시스템을 표현하는데 그래프가 가장 적절하고, 따라서 일반 시스템과 달리 위상적인 성질(topological property)이 중요하다는 것이다. 둘째는 시스템의 구성요소에 수송량(traffic)이 포함된다는 것이다.^[1,6] 최근 이와 같은 네트워크 시스템의 신뢰성을 계산하는 알고리즘에 관한 연구가 증대되고 있다.^[1,6,16,20] 가장 일반적인 신뢰성 척도는 k-terminal 신뢰성으로써 이는 네트워크에서 ver-

tex와 edge의 신뢰성이 주어질 때 k개의 특정 vertex가 서로 연결이 될 확률을 의미한다. 이러한 척도는 전통적인 시스템에서는 다루지 않던 척도로서 네트워크 시스템의 특성을 반영하는 것이다. 즉, 분권화된 네트워크(distributed network)의 경우 시스템의 성공(success)을 뜻하는 사건(event)은 우리가 어떤 vertex를 주목하느냐에 달려 있는 것이다.

그런데 실제 네트워크 시스템의 신뢰성은 이러한 다양한 사건(event)과 제약조건(가령 delay나 edge의 용량)에 달려 있다.^[16,18] 그러므로 위의 k-terminal 신뢰성 척도는 가장 단순한 척도라 할 수 있다. 이와 달리 현재까지 일반 시스템을 대상으로 한 신뢰성 모형과 제어론

*서울대학교 공과대학 산업공학과

**경기공업개방대학 산업공학과

† 이 논문은 1986년도 문교부 자유과제 학술연구조성비에 의하여 연구되었음.

(replacement poly, availability) 은 네트워크 시스템의 특성을 반영하지 못하고 있다.

최근 들어서 전통적인 신뢰성 이론과 개념을 시스템의 특성에 따라 심화, 확장시킨 연구가 수행되고 있으며, 일례로 통신네트워크에서 중요한 성능척도인 transit-delay나 edge (통신라인)의 용량을 제약조건으로 고려한 신뢰성 척도와 계산 알고리즘이 연구되고 있다.^[12,14,16] 본 논문은 이러한 연구의 일환으로 Birnbaum이 제시한 신뢰성 중요도라는 개념이 네트워크 시스템을 대상으로 어떻게 변형되어 네트워크의 특성을 제대로 고려한 개념이 될 수 있겠느냐의 문제를 다루고 있다. 이에 덧붙여 제시된 척도의 계산을 위해 최근에 개발된 네트워크 신뢰성의 계산 알고리즘의 적용의 문제도 다루고 있다.

따라서 본 논문에서 제시된 척도는 네트워크 시스템의 개발단계에서는 자원의 배분(resource allocation)에 관한 정보로 활용될 것이며, 보수단계에서도 우선순위에 관한 정보로 활용되는 실제적인 가치를 지닌 분석 도구가 될 것이다.

2. 성능을 고려한 요소의 신뢰성 중요도(Reliability Importance of the component considering performance)

기호

- V : vertex들의 집합 $\{v_1, \dots, v_m\}$
- E : vertex들의 쌍으로 정의되는 edge들의 집합 $\{e_1, \dots, e_n\}$
- $G(V, E)$: V와 E로 이루어진 그래프
- C : 구성요소(component)들의 집합 $\{C_1, \dots, C_n\}$
- s : 특정 vertex인 source
- t : 특정 vertex인 terminal
- P_i : edge e_i 의 신뢰성
- $N(G, P_i, s, t)$: 그래프 G와 edge들의 신뢰성, 그리고 source와 terminal로 이루어진 네트워크

$G * e$: 그래프 G에서 edge e를 제거하고 이에 인접한 두 vertex를 결합시킨 그래프

$G - e$: 그래프 G에서 edge e를 제거한 그래프

\parallel : $x_1 \parallel x_2 = 1 - (1 - x_1)(1 - x_2)$ 를 나타내는 기호(병렬구조를 나타내는 기호)

$R_{ij}(G)$: 그래프 G에서 vertex v_i 와 v_j 를 source와 terminal로 둘 때, source-to-terminal 신뢰성

$I_{ij}(ea)$: vertex v_i 와 v_j 를 source와 terminal로 둘 때 source-to-terminal 신뢰성에 대한 edge e_a 의 중요도

γ_{ij} : vertex v_i 와 v_j 간의 수송량(traffic)

가정

(1) 요소와 시스템의 상태는 작동과 작동중지의 두가지이다.

(2) 각 요소들은 서로 독립이다.

네트워크 시스템의 설계 문제는 vertex가 주어질 때 edge와 vertex의 incidence relation을 정하여 그래프의 connectivity나 연결 vertex쌍의 갯수를 최대화하는 문제를 많이 다루고 있다.^[6,8,10] 즉, 실제적으로 통신네트워크나 수송네트워크에서 computer center나 power generation center등은 location problem으로 정해지고 나서 이들간의 통신라인 혹은 수송라인을 정하는 문제로 단계화된다.^[8,10] 따라서 본 논문에서도 네트워크 시스템의 중요도를 따질 요소로서 edge를 대상으로 하는 것이 합당하다. 그런데 일반 시스템을 대상으로 한 신뢰성 중요도는 다음과 같이 주어진다.^[6]

$$I_h(C_i) = \frac{\partial h(p)}{\partial p_i} \dots \dots \dots (1)$$

식(1)은 요소 C_i 의 신뢰성을 Δ 만큼 증가시킬 경우 시스템의 신뢰성 $h(p)$ 가 증가되는 상대적인 비율을 나타낸다.

우리가 시스템의 신뢰성을 증가시키기 위해서

구성요소의 신뢰성을 증가시키고자 할 때, 식 (1)로 인해 우리는 개개의 요소들이 기여하는 바의 가중치가 바로 신뢰성 중요도임을 알 수 있다. 그런데 네트워크 시스템의 신뢰성 척도로 가장 일반적인 것은 k-terminal 신뢰성이다. 따라서 그래프 $G(V, E)$ 로 표현되는 네트워크시스템에서 $k=2$ 에서 $k=|V|$ 까지 서로 다른 시스템 신뢰성 척도의 총 경우의 수는

$$\binom{|V|}{2} + \binom{|V|}{3} + \dots + \binom{|V|}{|V|} = 2^{|V|} - |V| - 1$$

이다.

이 경우 주어진 네트워크에서 하나의 edge의 신뢰성을 증가시킬 때, 어느 vertex들의 집합이 연결될 vertex들의 집합으로 정해졌느냐에 따라, 그 edge의 신뢰성 중요도가 달라 나올 것이다. 이에 따라 한 edge에 대해, 신뢰성 중요도의 총 경우의 수는 $2^{|V|} - |V| - 1$ 이다. 따라서 우리가 특정 vertex에만 주목하지 않고, 네트워크 전체의 성능을 향상시키고자 할 때 $2^{|V|} - |V| - 1$ 의 신뢰성 중요도를 모두 고려해야 한다. 그러나 $|V|$ 가 10이상만 되어도 현실적으로 계산이 어려우므로 의미가 없다.

여기서 네트워크 시스템의 전반적 측면을 고려하는 방식으로 vertex들의 쌍(source-terminal pair)의 가능한 모든 집합을 생각할 수 있다. 즉, $\binom{|V|}{2}$ 개의 source-terminal쌍을 고려함으로써 네트워크 신뢰성의 전반적 측면을 고려하고 이에 따른 종합적인 신뢰성 중요도를 고찰하고자 한다. 이것은 모든 terminal이 한편으로 source가 되는 통신 네트워크나 컴퓨터 네트워크에서 특히 적합한 접근방법이라 할 수 있다. 이 경우 네트워크 시스템의 신뢰성 증가는 $\binom{|V|}{2}$ 개의 vertex 쌍에 대해 $R_{ij}(G)$ 를 증가시킴으로써 이루어진다. 이를 우리는 두가지 방법으로 모형화 할 수 있다.

- (1) $\binom{|V|}{2}$ 개의 목적함수를 가진 다목적함수(multi-objective function)의 종합화를 통한 중

합적인 신뢰성 중요도

- (2) 시스템 상태를 다상태(multi-state)로 정의하여^(6,11) 이에 따른 신뢰성 중요도

(2)의 경우에 있어 실제 평균교신 vertex 쌍의 숫자(expected number of communicated center pairs)에 관한 연구가 수행되는 것을 고려하면^(6,11) 교신 vertex쌍의 갯수에 따라 시스템 상태를 정의할 수 있다.

따라서 시스템 상태는 0에서 $\binom{|V|}{2}$ 까지의 값을 갖게 된다. 그러나 현재 적어도 k개의 vertex쌍이 교신할 확률을 계산하는 효율적인 알고리즘이 개발되지 않은 상태이므로 계산상의 문제가 존재한다.

(1)의 경우 다목적 함수를 종합화하는 여러가지 기법(Goal Programming이나 Fuzzy Programming등)이 있으나 우리는 가장 간단한 가중평균(weighted average)을 사용하고자 한다. 이때 가중치로는 vertex쌍간의 상호 주고받는 traffic양이 적합할 것이다. 따라서 그래프 $G(V, E)$ 와 traffic행렬($\{\gamma_{ij}\}$)로 이루어진 네트워크에서 traffic을 고려한 시스템의 종합적인 신뢰성은 다음과 같다.

$$T(G) = \sum_{i < j} \gamma_{ij} R_{ij}(G)$$

여기서 $T(G)$ 는 traffic을 고려한 시스템의 종합적인 신뢰성을 나타낸다. 이에따라 edge e_a 의 신뢰성 중요도를 정의하면 다음과 같다.

정의 1

그래프 G 와 edge e_i 의 신뢰성 p_i , 그리고 vertex간의 수송량 $\{\gamma_{ij}\}$ 가 주어졌을 때, edge e_a 의 종합적인 신뢰성 중요도의 척도는

$$I_G(e_a) = \frac{\sum_{i < j} \gamma_{ij} I_{ij}(e_a)}{\binom{|V|}{2}} \dots \dots \dots (2)$$

로 정의된다.

위의 척도의 특징은 다음과 같다. 첫째, 네트워크 시스템의 신뢰성 척도로 교신 vertex쌍의

평균값을 취하는 사실에서 볼 수 있듯이 실제 분권화된 네트워크에서 traffic을 주고 받는 것이 특정의 두 vertex라 할 때 가능한 모든 경우를 고려하였으므로 네트워크 시스템의 전반적 측면을 고려한 것이다. 둘째 실제 네트워크에서 특정 vertex가 시스템에서 차지하는 역할이 다르게 되고, (극단적인 경우로 centralized network의 center vertex를 고려해볼 수 있다) 따라서 이의 수량화가 vertex간의 traffic 양으로 나타난다면 이러한 사실을 반영할 것이라 할 수 있다.

3. 알고리즘 및 실험

3.1 알고리즘

2절에서 정의된 신뢰성 중요도의 척도인 식 (2)의 $I_c(e_a)$ 를 계산하기 위해서는 $I_{ij}(e_a) = \partial R_{ij}(G) / \partial P_a$ 를 먼저 계산해야 한다. 그런데 vertex 쌍의 총 갯수가 $\binom{|V|}{2} = |V|(|V| - 1) / 2$ 이므로 $\partial R_{ij}(G) / \partial P_a$ 를 계산할 때 exponential time이 걸릴 경우 이러한 척도는 계산시간(computation time)의 관점에서 유용성을 잃게 된다. 그러나 최근 특정 부류의 그래프에 대해서 polynomial-time 알고리즘이 개발되었으며,^[1, 2, 15, 20] 또한 pivotal decomposition이 네트워크 시스템에서 factoring이라는 성질로 표현되어 $\partial R_{ij}(G) / \partial P_a$ 의 계산을 용이하게 해주고 있다. 즉,

Proposition 1.

$$I_{ij}(e_a) = \frac{\partial R_{ij}(G)}{\partial P_a} = R_{ij}(G * e_a) - R_{ij}(G - e_a)$$

증명) 일반적인 coherent system에서의 pivotal decomposition을 네트워크에 적용해 보자.

$$\begin{aligned} R_{ij}(G) &= P_a R_{ij}(G/e_a : \text{작동}) + (1 - P_a) R_{ij}(G/e_a : \text{작동중지}) \\ &= P_a R_{ij}(G * e_a) + (1 - P_a) R_{ij}(G - e_a) \end{aligned}$$

그러므로

$$\frac{\partial R_{ij}(G)}{\partial P_a} = R_{ij}(G * e_a) - R_{ij}(G - e_a)$$

Proposition 1로부터 우리는 $I_c(e_a)$ 의 계산이 $\binom{|V|}{2}$ 개의 $G * e_a$ 와 $G - e_a$ 의 부분 그래프의 source-to-terminal 신뢰성 계산 문제가 됨을 알 수 있다. 그런데 이러한 척도($R_{ij}(G)$)에 대한 가장 효율적인 알고리즘은 최근에 개발된 polygon-to-chain reduction 기법을 이용하는 Factoring 알고리즘^[20]이다. Factoring 알고리즘을 서브루틴으로 사용한 $I_c(e_a)$ 의 계산 알고리즘은 다음과 같다.

입 력 : 네트워크 $N(G, P_a, e_a)$, traffic $\{\gamma_{ij}\}$

출 력 : $I_c(e_a)$

알고리즘

단계 0 : $i = 1, j = i + 1$

단계 1 : $G = (V, E)$ 와 e_a 로부터 $G_1 = G * e_a$ 와 $G_2 = G - e_a$ 를 구축한다.

단계 2 : $N_1 = (G_1, P_a, v_i, v_j)$ 에 대해 일반 네트워크 신뢰성 알고리즘을 적용하여 $R_{ij}(G_1)$ 을 구한다. $N_2 = (G_2, P_a, v_i, v_j)$ 에 대해서도 마찬가지로 $R_{ij}(G_2)$ 를 구한다.

단계 3 : 단계 2와 Proposition 1로부터 $I_{ij}(e_a) = R_{ij}(G_1) - R_{ij}(G_2)$ 를 계산한다.

단계 4 : $j < |V|$ 이면 $j = j + 1$ 로 두고 단계 2로 간다. 그렇지 않고 $|i| < |V| - 1$ 이면 $i = i + 1, j = i + 1$ 로 두고 단계 2로 간다. 그렇지 않으면

$$I_c(e_a) = \frac{\sum_{i,j} \gamma_{ij} I_{ij}(e_a)}{\binom{|V|}{2}}$$

를 계산한다.

이 알고리즘에 따른 $I_c(e_a)$ 를 계산하는 시간은 최악의 경우 exponential time이다. 이것은 G 가 일반그래프일 때 $R_{ij}(G)$ 의 계산문제가 NP-hard라는 사실^[4]에 기인한다. 그러나 G 가 특별한 그래프일 때는 polynomial time내에 $I_c(e_a)$ 의 계산이 가능하다. 이의 고찰을 위해 다음 Lemma 1을 보자.

Lemma 1.

네트워크를 이루는 그래프 G 가 직병렬 그래프(series-parallel graph)일 때 $R_{ij}(G)$ 의 계산

은 $O(|E|)$ 를 요구한다.
 증명) [1]에 나와 있다.

Proposition 2.

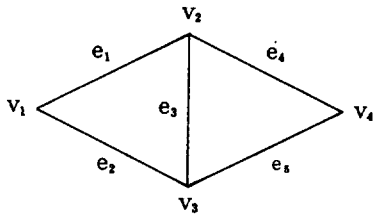
네트워크를 구성하는 그래프 G 가 다음 두 가지를 만족한다고 하자.

- (i) $G * e_a$ 가 직병렬 그래프이다.
- (ii) $G - e_a$ 가 직병렬 그래프이다. 그러면 $I_{ij}(e_a)$ 의 계산은 $O(|V|^2|E|)$ 를 요구한다.

증명) 알고리즘에서 생성되는 서로 다른 부분그래프 G_1 과 G_2 의 총수는 최악의 경우 $\binom{|V|}{2} \cdot 2 = |V|(|V| - 1)$ 이다. 따라서 알고리즘은 $|V|(|V| - 1)$ 번의 Factoring 알고리즘을 수행한다. 그런데 Lemma 1로부터 각 Factoring 알고리즘은 $O(|E|)$ 의 시간이 소요된다. 그러므로 총소요시간은 $O(|V|(|V| - 1)|E|) = O(|V|^2|E|)$ 이다.

3.2 실례

그림 1과 같은 네트워크와 traffic을 고려해 보자.



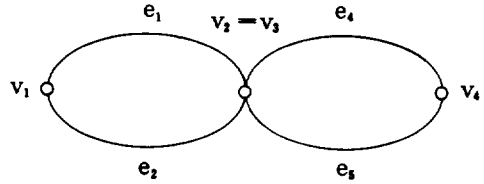
$$[\gamma_{ij}] = \begin{matrix} & v_1 & v_2 & v_3 & v_4 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{matrix} & \begin{bmatrix} 0 & 1 & 0.8 & 0.6 \\ 1 & 0 & 0.7 & 0.7 \\ 0.8 & 0.7 & 0 & 0.8 \\ 0.6 & 0.7 & 0.8 & 0 \end{bmatrix} \end{matrix}$$

[그림 1] 실례 네트워크와 traffic

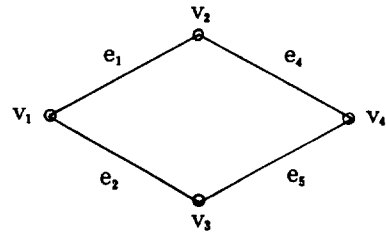
여기서 vertex v_i 와 v_j 의 수송량은 서로가 source와 terminal이 되어 주고받는 총량을 0에서 1사이의 값을 갖도록 보정해 준 값이다.

그리고 $\gamma_{ij} = \gamma_{ji}$ 이다.

edge e_3 를 택해 $I(e_3)$ 를 구해보자. 단계 1에 의한 $G * e_3$ 가 그림 2에 나와 있다.



$G * e_3$



$G - e_3$

[그림 2] $G * e_3$ 와 $G - e_3$

따라서 $\binom{4}{2} = 6$ 개의 source-terminal 쌍에 관해 간단한 series-parallel reduction을 사용한 단계 2를 적용하면 $R_{ij}(G_1)$ 과 $R_{ij}(G_2)$ 는 P_1, P_2, P_4, P_5 를 이용해 표 1과 같이 구해진다.

[표 1] G_1 과 G_2 에서 모든 source-terminal 쌍에 대한 $R_{ij}(G)$

$R_{12}(G_1) = P_1 \parallel P_2$	$R_{12}(G_2) = P_1 \parallel (P_2 P_4 P_5)$
$R_{13}(G_1) = P_1 \parallel P_2$	$R_{13}(G_2) = P_2 \parallel (P_1 P_4 P_5)$
$P_{14}(G_1) = (P_1 \parallel P_2) \cdot (P_4 \parallel P_5)$	$R_{14}(G_2) = P_1 P_4 \parallel P_2 P_5$
$R_{23}(G_1) = 1$	$R_{23}(G_2) = P_1 P_2 \parallel P_4 P_5$
$R_{24}(G_1) = P_4 \parallel P_5$	$R_{24}(G_2) = P_4 \parallel P_1 P_2 P_5$
$R_{34}(G_1) = P_4 \parallel P_5$	$R_{34}(G_2) = P_5 \parallel P_1 P_2 P_4$

$P = (P_1, P_2, P_3, P_4, P_5)^T = (0.9, 0.9, 0.95, 0.8, 0.9)$ 라 할 때 단계 2에서 구해진 표 1을 토대로 단계 3을 적용하면,

$$I_{12}(e_3) = 0.0252, \quad I_{23}(e_3) = 0.0532$$

$$I_{13}(e_3) = 0.0252, \quad I_{24}(e_3) = 0.0342$$

$$I_{14}(e_3) = 0.0234, \quad I_{34}(e_3) = 0.0152$$

가 된다. 따라서 단계 4에 의해

$$I_G(e_3) = \frac{1}{6} [1 \times 0.0252 + 0.8 \times 0.0252 \\ + 0.6 \times 0.0234 + 0.7 \times 0.0532 \\ + 0.7 \times 0.0342 + 0.8 \times 0.0152] \\ = 0.02212$$

가 된다.

4. 결론 및 추후 연구사항

본 논문에서는 네트워크 시스템을 대상으로 하여 traffic과 신뢰성을 고려하여 edge의 중요도에 관한 척도를 제시하고 이의 계산에 관해 논해 보았다. 현재는 네트워크의 일반적인 신뢰성 척도인 k-terminal 신뢰성 문제가 NP-hard^[6,11]임에 따라 특정 구조를 지닌 부분 네트워크에 대해 polynomial time 알고리즘을 개발하려는 노력이 집중되었다. 이의 결과에 따라 실제의 ARPA(Advanced Research Project Agency) - 네트워크를 포함하는 cube-free 그래프에 대해서 all-terminal 신뢰성을 계산하는 polynomial-time 알고리즘이 Satyanarayana가 Politof에 의해 개발되었다. 그런데 k-terminal 신뢰성의 경우 실제 네트워크 시스템의 신뢰성 모형에 있어 delay나 traffic 등의 제약조건과 edge간의 dependence 등을 고려치 않은 가장 간단한 척도이다. 따라서 이러한 간단한 척도를 계산하는 알고리즘이 발전됨에 따라 위의 제약조건을 이루는 네트워크의 요소를 고려한 척도를 제시하고자 하는 연구가 진행되고 있으며^[12,13] 본 논문의 traffic을 고려한 신뢰성 중요도의 척도도 이러한 추세를 반영한 것이라 할 수 있겠다.

본 논문에서 제시된 척도의 특징은 다음과 같다. 첫째, 네트워크 시스템에서 시스템의 작동(success)은 vertex 쌍 간의 연결을 의미하고 있으며,^[11] 본 논문의 척도는 이러한 사실을 반영하고 있다. 둘째, 컴퓨터가 시스템에 도입된 이후 고도의 redundancy가 도입되어 시스템 자체에 완전 중지(failure)는 거의 일어나지 않으므로 전통적인 신뢰성은 1에 가까워지고, MT-TF(Mean time to failure) 역시 그 의미를 잃고 있으므로 traffic을 고려함으로써 이러한 현상을 상쇄하였다. 그리고 각 vertex의 중요도가 상이할 경우 이러한 것이 traffic을 통해 반영된다 하겠다.

끝으로 여기서 좀더 연구되어야 할 점 두가지만 지적하고자 한다. 첫째, 네트워크 시스템에 있어서 특정 vertex v_i 와 v_j 를 주목할 때 이들 사이의 연결 상태가 0(연결 안됨)과 1(연결) 두가지 상태로 규정짓는 것이 통례이다.

그런데 실제로는 연결이 가능한 상태를 두 vertex간에 서로 다른 몇 개의 path가 존재하느냐로 세분할 수 있다. 이에 따라 네트워크에서의 다상태(multi state) 신뢰성 모형을 고려해 볼 수 있다. 그리고 그러한 다상태 신뢰성 모형의 신뢰성 중요도를 Buttler, Natvig, 그리고 A. P. Wood가 제시한 척도에 따라 다양하게 고려해 볼 수 있다. 그러나 이 경우 두 vertex가 주어질 때 적어도 k개의 path가 작동할 확률을 계산하는 효과적인 알고리즘에 대한 연구가 선행되어야 한다.

둘째, 본 논문에서는 vertex 쌍 간의 traffic만을 고려했으나 실제로 이러한 traffic이 routing 알고리즘과 시간에 따라 변하면서 특정 edge를 사용하여 운송되고 있다. 따라서 이런 routing현상을 신뢰성 모형과 결합시켜 신뢰성 중요도를 고려해 볼 수도 있겠다.

References

1. Agrawal, A. and A. Satyanarayana (1985), "Network Reliability Analysis Using 2-Connected Digraph Reductions" Networks, Vol. 15, pp. 239-256.
2. Agrawal, A. and R.E. Barlow (1982), "A Survey of Network Reliability" ORC 83-5 Operations Research Center, Univ. of California, Berkeley.
3. Ball, M.O. (1979), "Computing Network Reliability" Operations Research Vol. 27, pp. 823-838.
4. Ball, M.O. (1986), "Computational Complexity of Network Reliability Analysis: An Overview" IEEE Trans. on Reliability, Vol. R-35, pp. 230-239.
5. Barlow R.E. and A.S. Wu (1978), "Coherent Systems with Multistate Components", Mathematics of Operations Research, Vol. 4, pp. 275-281.
6. Barlow, R.E. and F. Proschan (1981), "Statistical Theory of Reliability and Life Testing, Holt, Rinehart and Winston, INC.
7. Barlow, R.E. and F. Proschan (1975), "Importance of System Components and Fault Tree Events," Stochastic Process and Their Applications, Vol. 3, pp. 153-173.
8. Bauer, D., F. Proschan, C. Suffel and R. Tindel (1985), "Combinational Optimization Problems in the Analysis and Design of Probabilistic Networks," Networks, Vol. 15, pp. 257-271.
9. Boesch, F.T. (1986), "On Uncertainty Polynomials and Graph Connectivity in Reliable Network Synthesis," Journal of Graph Theory, Vol. 10, pp. 339-352.
10. Boesch, F.T. (1986), "Synthesis of Reliable Networks-A Survey," IEEE Trans. on Reliability, R-35, pp. 240-246.
11. Clark, B.N., E.M. Neufeld and C.J. Colbourn (1986), "Maximizing the Mean Number of Communicating Vertex Pairs in the Series-Parallel Networks", IEEE Trans. on Reliability, R-35, pp. 247-251.
12. Frank, H. and I. Frisch (1971) Communication, Transmissions and Transportation Networks, Addison-Wesley.
13. Griffith, W.S. (1980), "Multistate Reliability Models", J. Appl. Prob., Vol. 17, pp. 735-744.
14. Park, J.W. and G.M. Clark (1986), "A Computational Algorithm for Reliability Bounds in Probabilistic Design", IEEE Trans. on Reliability, R-35, pp. 30-31.
15. Politof, T. and A. Satyanarayana (1986), "Efficient Algorithms for Reliability Analysis of Planor Networks-A Survey", IEEE Trans. on Reliability, R-35, pp. 252-259.
16. Raghavendra, C.S. and S.V. Makam (1986), "Reliability Modeling and Analysis of Computer Networks", IEEE Trans. on Reliability, R-35, pp. 156-160.
17. Satyanarayana, A. (1982), "A Unified Formula for Analysis of some Reliability Problems", IEEE Trans. on Reliability, R-35, pp. 23-32.
18. Tanenbaum, A.S. (1981), Computer Networks, Prentice-Hall.
19. Valiant, L.G. (1979), "The Complexity of Enumeration and Reliability Problems", SIAM J. Compute, Vol. 8, pp. 418-421.
20. Wood, R.K. (1986), "Factoring Algorithms for Computing K-Terminal Network Reliability", IEEE Trans. on Reliability, R-35, pp. 269-278.