

# Fault-tree 확률계산을 위한 근사적 방법

## An approximate approach for Computing Fault-tree probabilities

李 根 熙\*  
李 東 炯\*\*

### ABSTRACT

This paper presents an approximate algorithm for computing Fault-tree probabilities. The method is essentially composed of three steps.

In the first step, a Fault-tree is converted into a network form.

In the second step, We change the network into a parallelized diagram.

In the third step, the approximate fault-tree probability is calculated from the parallelized diagram.

In this paper, in order to verify the method two hypothetical Fault-tree is used by examples.

The results show that the method is very useful, even though it is an approximate technique, since it needs not to search the minimal cut sets and has the simple computing routines.

### 1. 서 론

Fault tree(이하 FT) 분석은 복잡한 system

의 신뢰도를 모델링하고 계산하는데 매우 강력하고 효과적인 기법이다. 다시 말하면 FTA (Fault Tree Analysis)는 system고장을 유발시

\* 한양대학교 산업공학과  
\*\*대전공업대학 산업공학과

키는데 영향력 있는 여러가지 요인들의 관계를 Boolean Logic gate를 이용, 도해적으로 표현하여 분석하는 기법이다.

따라서 Fault trees 혹은 Event trees는 Top event와 Top event를 발생시킬수 있는 경로들을 나타낸 logic diagrams이다. Fault trees는 Top event가 system 고장일때 사용되는 반면 Event trees는 system 성공일 경우에 사용된다.

그 트리구조는 신뢰도 block diagram에서의 직·병렬구조와 같이 똑같은 기능을 수행하는 AND gates와 OR gates로 구성되어 있다.

Fault tree의 분석에서 주요한 단계는 minimal cut sets을 구하는 것이다. minimal cut sets이란 cut sets중 중복되는 집합을 제거한 후 남은 cut sets으로서 Top사상을 발생하는데 최소한의 cut sets을 의미하는데 지금까지 이를 구하는 방법이 많이 나와있다. [4]

Minimal cut sets이 구해지면 이를 근거로 Fault tree(Top event) 확률을 계산하는데 방법으로는 신뢰도 블록 다이어그램으로 전환하여 푸는 방법, Inclusion-Exclusion(IE)방법, SDP(Sum of disjoint product)방법등이 있다.

그러나 IE방법과 SDP방법은 기본사상의 수가 증가함에 따라 지수적으로 계산량의 증가를 가져오기 때문에 기본사상이 많은 경우 실제로 적용하기가 힘들다.

최근들어 매우 정밀한 정확도를 요구하지 않는 system의 FT확률을 계산하는데 있어서 정확한 값을 구하는데 소요되는 시간과 노력문제를 감안, 근사적 해법의 개발이 점차 그 중요성을 더해가고 있다. 특히 FT에서 좋은 보전활동과 짧은 사용기간 때문에 모든 부품의 고장확률이 매우 작을 경우에는 근사값도 좋은 지표로 활용될 수 있다. 또한 혼란상태의 운용조건(예, 전쟁터에서 사용)을 가졌거나 장기간 사용(우주

공간에서 사용)하는 경우에는 이런 방법은 비교적 손쉽게 바람직한 결론을 도출할 수 있다. [7]

2장에서는 기존의 FT의 확률계산 방법중 본고의 주제와 가장 관련이 깊은 SDP법에 대해 설명하고 3장에서는 근사적 해법을 제시하고자 한다.

## 2. SDP법

### 2-1 SDP법의 개요

FT의 확률과 네트워크 신뢰도를 계산하는 해법에는 IE법, SDP법, 분해법, 위상대수법등 여러 범주로 나눌 수 있지만 여기서는 SDP법에 대해서만 간략히 설명한다.

SDP법은 IE법과 마찬가지로 정해진 경로에 대해 계속적으로 배반인 곱의 항을 만든뒤 이를 합산해 나가는 방법이다.

SDP법은  $P_r\{P_1 \cup P_2\} = P_r\{P_1\} + P_r\{\bar{P}_1 \cap P_2\}$  이라는 법칙을 이용, system 확률을 도출해 낸다. (여기서  $\bar{P}_1$ 은  $P_1$ 의 여사상 즉  $P_r\{\bar{P}_1\} = 1 - P_r\{P_1\}$  임)

따라서

$$\begin{aligned}
 R &= P_r\left(\bigcup_{i=1}^r A_i\right) \\
 &= P_r\{A_1\} + P_r\{\bar{A}_1 \cap A_2\} \\
 &\quad + P_r\{\bar{A}_1 \cap \bar{A}_2 \cap A_3\} + \dots \\
 &\quad \dots P_r\{\bar{A}_1 \cap \bar{A}_2 \cap \dots \cap \bar{A}_{r-1} \cap A_r\} \\
 &= P_r(A_1) + P_r(\bar{A}_1 A_2) + P_r(\bar{A}_1 \bar{A}_2 A_3) + \dots \\
 &\quad + P_r(\bar{A}_1 \bar{A}_2 \dots \bar{A}_{r-1} A_r) \\
 &\quad \dots \dots \dots (2-1)
 \end{aligned}$$

되는데 +, -부호를 사용하는 IE법과 달리 +부호만으로 처리되는 것이 특징이다.

식 (2-1)에서의 항의 총 수는  $r$ 이나 각항의 계산은 지수시간(exponential time)을 요구한다.

(2-1)에 근거하여  $r$ 을 계산하는 SDP법은 통상 Boolean Algebra이용 방법으로 지칭되기도 하며 그동안 많은 연구가 있었다. [3, 4, 5, 6] 이 방법은 같은 부품이 서로 다른 경로에 중복되어 포함되는 경우, 각각의 경로들 중에서 중복 계산되는 부품들을 어떻게 효율적으로 처리할 것인가 하는 방법상의 문제가 있다.

1979년 Abraham[1]의 기존의 방법보다 효율적인 방법을 개발한 이래 최근까지 더 개선된 알고리즘들이 많이 제시되고 있다.

2-2. 계산예

SDP법을 설명하기 위해 <그림 1>과 같은 FT를 예로 들어보자. (각 기본사상의 고장확률은 0.1이라 함)

먼저 minimal cut sets을 구하면  $x1x3, x2x3, x6, x4x5$ 이다. 이들을 항목의 수가 적은 것 부터 배열하면

Minimal cut sets =  $x6 + x1x3 + x2x3 + x4x5$ 가 되는데 이 항들을 배반되게 하기 위해 다음과 같이 SDP법을 적용한다.

1단계 : 첫 항으로  $x6$ 을 선택하고 식 (2-1)과 같이 전개한다. 이때 전개된 항들중 중복된 항이 있으면 제거한다.

결과 :  $x6 + \overline{x6}x1x3 + \overline{x6}x2x3 + \overline{x6}x4x5$

2단계 : 둘째항으로  $\overline{x6}x1x3$ 을 선택하고 1단계와 같이 한다.

결과 :  $x6 + \overline{x6}x1x3 + \overline{x6}\overline{x1}x2x3 + \overline{x6}\overline{x1}x4x5 + \overline{x6}x1\overline{x3}x4x5$

3단계 : 세째항으로  $\overline{x6}\overline{x1}x2x3$ 을 선택하고 1단계와 같이 한다.

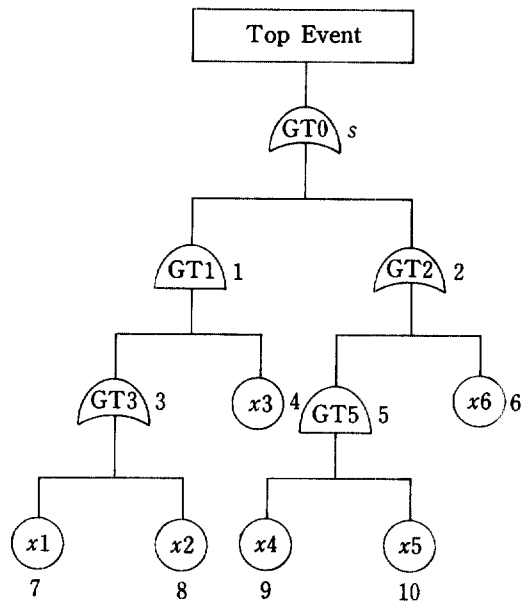
결과 :  $x6 + \overline{x6}x1x3 + \overline{x6}\overline{x1}x2x3 + \overline{x6}x1\overline{x3}x4$

$x5 + \overline{x6}\overline{x1}x2x4x5 + \overline{x6}\overline{x1}x2\overline{x3}x4x5$

4단계 : 네째항으로  $\overline{x6}x1\overline{x3}x4x5$ 을 선택하고 1단계와 같이 한다.

결과 :  $x6 + \overline{x6}x1x3 + \overline{x6}\overline{x1}x2x3 + \overline{x6}x1\overline{x3}x4x5 + \overline{x6}\overline{x1}x2x4x5 + \overline{x6}\overline{x1}x2\overline{x3}x4x5$

5단계 : 모든 항이 배반이 되었기 때문에 각 항에 직접 확률값을 대입, FT확률  $P(F)$ 를 계산하면  $P(F) = 0.125929$ 가 된다.



<그림 1> 단순한 Fault tree

다음에는 [2]에서 인용된 <그림 2>와 같은 반복사상이 있는 F.T를 예로 들어보자.

우선 minimal cut sets을 구하면 다음과 같다.

- ED, EGH, EFH, LKD, LKGH, LKFG,
- LKFH, JLD, JLGH, JLFG, JLFH, JKD,
- JKGH, JKGH, JKFG, JFKH

(1) 가장 작은 항목인  $ED$ 를  $A_1$ 로 하여 식 (2-1)에 의해 전개한뒤 중복항을 제거하면 다음과 같다.

$$\begin{aligned}
 &ED + \bar{E}\bar{D}GH + \bar{E}\bar{D}FG + \bar{E}\bar{D}FH \\
 &+ \bar{E}\bar{D}LK + \bar{E}\bar{L}KGH + \bar{E}\bar{D}LKGH \\
 &+ \bar{E}\bar{L}KFG + \bar{E}\bar{D}LKFG + \bar{E}\bar{L}KFH \\
 &+ \bar{E}\bar{D}LKFH + \bar{E}\bar{J}LD + \bar{E}\bar{J}LGH \\
 &+ \bar{E}\bar{D}\bar{J}LGH + \bar{E}\bar{J}LFG + \bar{E}\bar{D}\bar{J}LFG \\
 &+ \bar{E}\bar{J}LFH + \bar{E}\bar{D}\bar{J}LFH + \bar{E}\bar{J}KD \\
 &+ \bar{E}\bar{J}KGH + \bar{E}\bar{D}\bar{J}KGH + \bar{E}\bar{J}KFG \\
 &+ \bar{E}\bar{D}\bar{J}KFG + \bar{E}\bar{J}KFH + \bar{E}\bar{D}\bar{J}KFH
 \end{aligned}$$

(2) 다음에는  $\bar{E}\bar{D}GH$ 를  $A_1$ 를 하여 (1)과 같이 하면 다음과 같다.

$$\begin{aligned}
 &ED + \bar{E}\bar{D}GH + \bar{E}\bar{D}G\bar{H}F + \bar{E}\bar{D}\bar{G}\bar{H}F \\
 &+ \bar{E}\bar{L}KD + \bar{E}\bar{L}KGH + \bar{E}\bar{L}KFH \\
 &+ \bar{E}\bar{J}LD + \bar{E}\bar{J}LGH + \bar{E}\bar{J}LFG \\
 &+ \bar{E}\bar{J}LFH + \bar{E}\bar{J}KD + \bar{E}\bar{J}KGH \\
 &+ \bar{E}\bar{J}KFG + \bar{E}\bar{J}KFH
 \end{aligned}$$

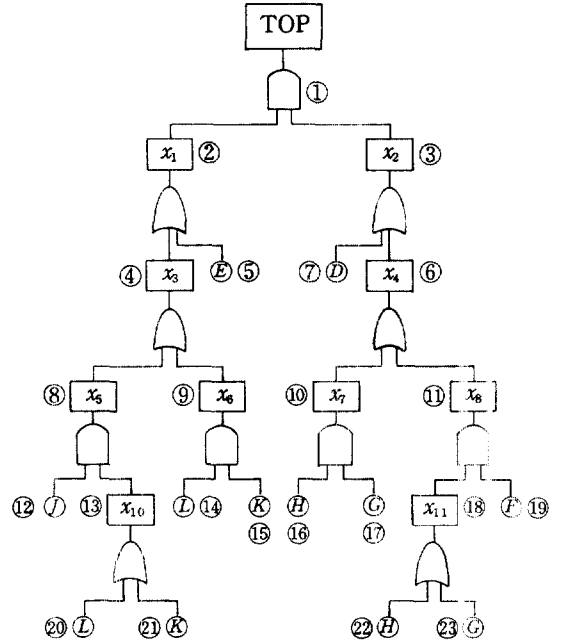
(3) 이번에는  $\bar{E}\bar{L}KD$ 를  $A_1$ 로 하고 전단계와 같이 한다.

.....(중간계산생략)

(15) 이와 같은 방법으로 계속 반복하게 되면 최종단계에서 다음과 같은 배반항들의 집합을 구할 수 있다.

$$\begin{aligned}
 &ED + \bar{E}\bar{D}GH + \bar{E}\bar{L}KD + \bar{E}\bar{D}G\bar{H}F \\
 &+ \bar{E}\bar{D}F\bar{G}\bar{H} + \bar{E}\bar{L}K\bar{D}\bar{J} + \bar{E}\bar{D}\bar{J}\bar{L}K \\
 &+ \bar{E}\bar{L}K\bar{D}\bar{G}\bar{H} + \bar{E}\bar{L}\bar{K}\bar{J}G\bar{H} + \bar{E}\bar{L}K\bar{D}\bar{G}\bar{H}F \\
 &+ \bar{E}\bar{L}\bar{D}\bar{K}\bar{G}\bar{H}F + \bar{E}\bar{K}\bar{L}\bar{J}\bar{D}\bar{G}\bar{H} + \bar{E}\bar{L}K\bar{D}\bar{J}F\bar{H} \\
 &+ \bar{E}\bar{L}\bar{K}\bar{J}\bar{D}\bar{G}\bar{H}F + \bar{E}\bar{L}\bar{K}\bar{J}\bar{G}\bar{H}\bar{D}F \\
 &+ \bar{E}\bar{K}\bar{L}\bar{J}\bar{D}\bar{G}\bar{H}F
 \end{aligned}$$

따라서 각 기본사상의 고장확률이 0.1이라 할 때  $FT$  확률은  $P(F) = 0.0156183$ 이 된다.



<그림 2> 반복사상이 있는 F.T

### 3. 근사적 계산방법

#### 3-1. 가정

1) 모든 부품들 사이의 고장은 통계적으로 독립 ( $S$ -independent)이다.

2) FT의 기본사상의 확률은 알고 있다

#### 3-2. 알고리즘

단계 0: FT를 작성한다.

단계 1: FT를  $S$ - $T$  네트워크로 전환시킨다.

(i) FT의 각 gate와 기본사상에 대해 순차적으로 번호를 부여한다. (가상의 Terminal node는 마지막 번호가 됨)

(ii) AND gate는 직렬로, OR gate는 병렬로 작성한다.

(iii) 각 edge는 확률값을 가지는데 gate에서 gate로 가는 경우는 모든 확률을 1로 한다.

단계 2: 단계 1에서 작성된 네트워크를 병렬화 diagram으로 변형시킨다. (이때 확률 1인 edges는 하나의 연결선으로 간주함)

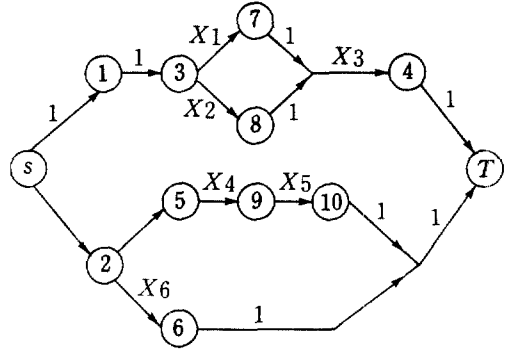
단계 3: 병렬화된 각각의 경로  $T_1, T_2, \dots$ 의 확률을 구한뒤 FT확률을 계산한다.

(i) 하위 병렬화된 부분은 식(2-1)인 외부순환식을 써서 서로 배반사상인 항들의 합으로 표시한다.

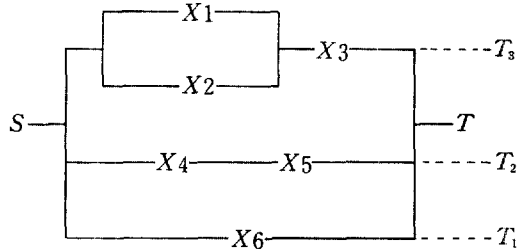
(ii) 반복되는 항목은 아이덴티티(예,  $AA=0, AA=A^2=A$ )법칙을 써서 제거한다.

(iii) 하위 경로가 직렬인 경우는 단순히 곱한다.

(iv) (iii)항이 있고 반복사상이 있으면 내부순환식을 써서 계산한다. 여기서 내부순환식을 경로  $A$ 가 2개의 구성부품  $a_1, a_2$ 로 구성되어 있을 때  $\bar{A} = (\overline{a_1 a_2}) = \bar{a}_1 + a_1 + \bar{a}_2$ 가 되는 식을 말한다.



<그림 3> (그림 1)의 네트워크화



<그림 4> (그림 3)의 병렬화 다이어그램

3-3. 계상에 및 SDP법과의 비교

먼저 본 근사해법을 설명하기 위해 <그림 1>을 예로 들어보자.

단계 1: 먼저 <그림 1>을 network 형태로 만들면 <그림 3>과 같다.

단계 2: 네트워크가 완성되면 <그림 4>와 같은 병렬화 다이어그램을 작성한다.

단계 3: 따라서 FT확률은 <그림 4>를 계산함으로써 구해질 수 있다.

즉  $T_1 = X_6, T_2 = X_4 + X_5, T_3 = (X_1 + \bar{X}_1 X_2) X_3$ 이므로

$$\begin{aligned}
 P(F) &= X_6 + \overline{X_6 X_4 X_5} (\overline{X_6 + \bar{X}_6 X_4 X_5}) \\
 &= (X_1 + \bar{X}_1 X_2) X_3 \text{ (반복사상이 없으므로 직접 계산)} \\
 &= 0.125929 \text{ (각 부품의 고장률은 0.1임)}
 \end{aligned}$$

여기서 구한 FT 확률값은 2장에서 SDP법으로 구한 결과와 같지만 기본사상이 반복된 경우에는 약간의 오차가 있다. 그러나 계산방법이 SDP법보다 매우 간단하다는 것을 알 수 있다. 즉, 최종 식이 나올때까지 반복여부를 확인하는

비교회수를 고려하지 않더라도 계산되는 곱의 항의 수는 SDP법이 11개인 반면 본근사해법은 4개항에 불과하다.

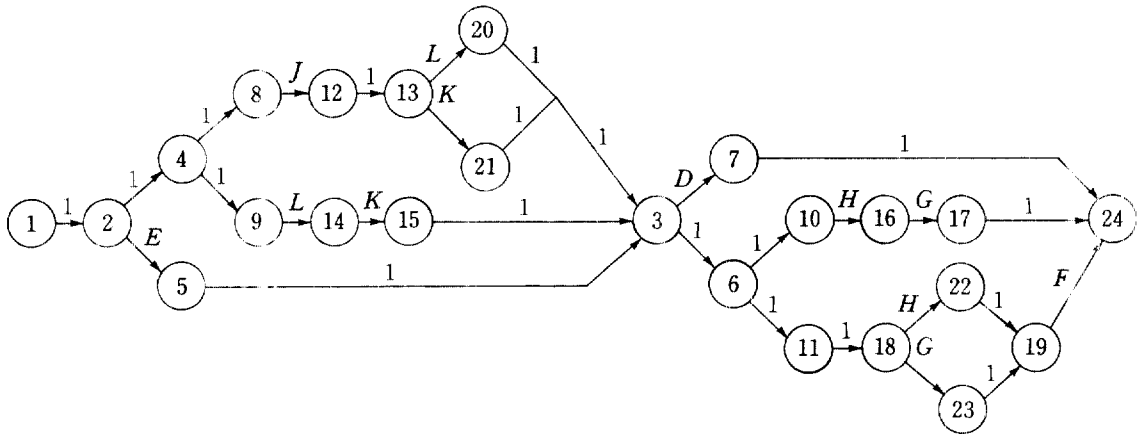
다음에는 앞에서 인용된 <그림 2>와 같은 반복사상이 있는 F.T에 적용해 보자.

단계 1. FT의 gate 및 기본사상에 1번부터 24

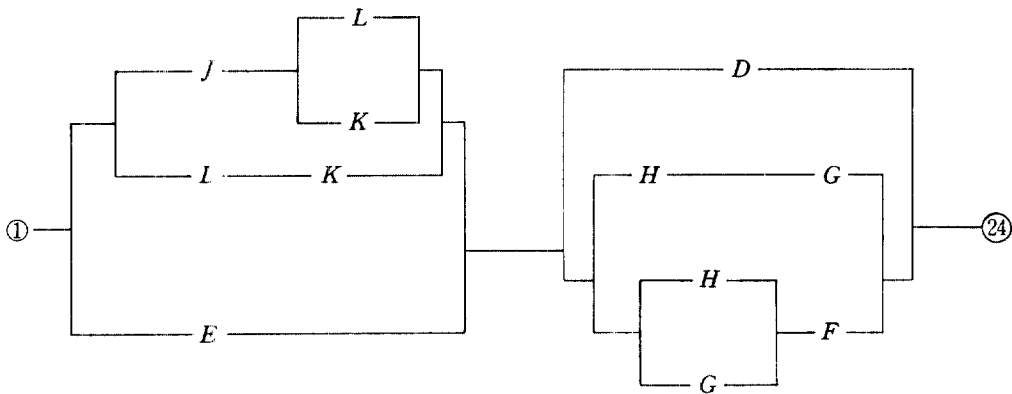
번 까지 번호를 부여하고 네트워크 형태로 전환한다. (그림 5 참조)

단계 2. (그림 5)을 병렬화 다이어그램으로 변형시킨다. (그림 6 참조)

단계 3. (그림 6)의 병렬화 다이어그램의 확률을 계산한다. (각 부품의 고장율 0.1)



<그림 5> (그림 2)의 네트워크화



<그림 6> (그림 5)의 병렬화 다이어그램

$$\begin{aligned}
 R &= [E + \{LK + \bar{L}\bar{K}(L + \bar{L}K)J\} E] \\
 &\quad \cdot [D + \{HG + \bar{H}\bar{G}(H + \bar{H}G)F\} \cdot \bar{D}] \\
 &= [E + \{LK + (\bar{L} + \bar{L}\bar{K})(L + \bar{L}K)J\} E] \\
 &\quad [D + \{HG + (\bar{H} + \bar{H}\bar{G}) \\
 &\quad (H + \bar{H}G)F\} \bar{D}] \\
 &= [E + \{LK + (\bar{L}K + \bar{L}\bar{K})J\} E] \\
 &\quad [D + \{HG + (\bar{H}\bar{G} + \bar{H}\bar{G})F\} \bar{D}] \\
 &= 0.01567504
 \end{aligned}$$

이 값은 앞에서 구한 SDP법보다 0.00005661만큼 더 큰 값을 나타내는데 이런정도의 차는 무시할만하여 고장률이 0.1보다 작은 경우에는 그 차이가 더욱 미미한 것으로 나타났다. (고장율이 0.09인 경우 차는 0.00003962임)

앞에서 두가지 예를 가지고 두 방법을 비교해 본 결과 본근사해법은 기본사상이 반복된 경우 SDP법의 계산 결과와 거의 무시할만한 오차를 나타냈지만 SDP법보다 계산 및 비교의 회수가 매우 적기 때문에 계산시간을 많이 단축시킬 수 있었다.

따라서 본근사방법은 일상의 FT확률문제의 유용한 계산방법으로 활용가능하다고 판단된

다.

#### 4. 결 론

본논문에서 제시한 FT확률의 근사적 계산방법은 FT를 network형태로 전환한 뒤 이를 병렬화하는 방법을 사용함으로써 예제에 적용해 본 결과 기존의 방법보다 계산절차가 단순하고 계산시간을 많이 단축시키는 것으로 나타났다.

따라서 본 방법은 소형계산기에 의한 수작업으로도 중규모정도의 반복사상이 있는 FT의 확률계산에 적용가능 하며 타 방법에 비해 계산절차가 매우 간단하다는 점에서 근사값이란 약점을 충분히 상쇄하고도 남을 것이다.

그러나 여기서의 해법이 FT는 물론 모든 형태의 네트워크에 적용 가능할 것인지는 하나의 문제점으로 지적될 수 있다. 가장 단순한 형태인 단방향 S-T 신뢰성 문제에는 적용이 가능할 것이나 무방향 네트워크에 대해서는 적용이 어려울 것으로 생각되는데 앞으로 이런 방향에서 연구가 있어야 할 것이다.

### 參 考 文 獻

1. Abraham, J. A. (1979), "*An Improved Alogrithm for Network Reliability*", IEEE Trans. Rel., Vol. R-28, No. 1, pp. 58-61.
2. Bengiamin, N. N., Bowen, B. A., Schenk, K. F. (1976), "*An Efficient Algorithm for Reducing The Complexity of Computation in Fault Tree Analysis*", IEEE Trans. on Nuclear Science, Vol. NS-23, No. 5, pp. 1442-1446.
3. Bennetts, R. G. (1982), "*Analysis of Reliability Block Diagrams by Boolean Technique*", IEEE Trans. Rel., Vol. R-31, No. 2, pp. 159-166.
4. Lee, W. S., Grosh, D. L., Tillman, F. A., Lie, C. H. (1985), "*Fault Tree Analysis, Method and Applications-A Review*", IEEE Trans. on Rel., Vol. R-34, No. 3, pp. 194-203.
5. Lin, P. M., Leon, B. J. and Huang, T. C. (1976), "*A New Algorithm for Symbolic System Reliability Analysis*", IEEE Trans. Rel., Vol. R-25, No. 1, pp. 2-15
6. Locks, M. O. (1985), "*Recent Developments in Computing of System Reliability*", IEEE Trans. Rel., Vol. R-34, No. 4, pp. 425-436.
7. Wood, A. P. (1985), "*Multistate Boick Diagrams and Fault Trees*" IEEE Trans. on Reliability, Vol. R-34, No. 3, pp. 236-240.