

해석적 방법에 의한 고장 수목 순환 논리의 분석: 실제 PSA에의 적용 예

양준언, 황미정, 한상훈, 김태운

한국 원자력 연구소

요 약

1단계 확률론적 안전성 평가 (Level 1 Probabilistic Safety Assessment, PSA)를 수행할 때 나타나는 보조계통 고장 수목간의 순환 논리는 사고 경위 정량화를 위하여 해결되어야만 한다. 기존의 PSA에서는 이를 위하여 별도의 고장 수목을 다시 작성하였으나, 이 방법을 사용하기 위하여서는 보조계통 간의 관계를 검토하여야 하며, 이에 따른 별도의 고장 수목을 작성하여야 하는 등 추가적인 작업이 요구된다. 또한 기존 방법은 일부 최소 단절군이 생성되지 않는 약점을 갖고 있다. 이에 따라 한국원자력연구소에서는 해석적으로 순환 논리를 푸는 방법을 개발하였으며, 이를 PSA용 코드인 KIRAP 코드에 구축하였다. 이에 따라 기존 방법의 약점을 극복하고 고장 수목간의 순환 논리를 자동으로 풀 수 있게 되었다. 본 논문에서는 개발된 해석적 방법을 설명하며, 또한 이 방법을 실제 PSA에 적용하며 나타난 여러 현상에 대하여 살펴본다.

1. 서 론

1단계 확률론적 안전성 평가 (Level 1 Probabilistic Safety Assessment, PSA)는 발전소의 설계, 운전 및 보수 등을 포함하는 발전소의 종합적 안전성을 평가하는 방법이다. 1단계 PSA에서는 이를 위하여 사건 수목 (Event Tree)과 고장 수목 (Fault Tree)을 사용한다. 사건 수목은 어떤 초기 사건에 대한 발전소 안전계통의 이용가능 여부에 따른 사고 경위를 보여주는 일종의 논리도이며, 고장 수목은 사건 수목에서 그 이용 가능성 여부를 묻는 발전소 안전계통의 이용 불능도를 계산하는 데 사용되는 것으로 AND, OR의 게이트 및 기본 사건 (Basic Event) 등으로 구성된 논리도이다.

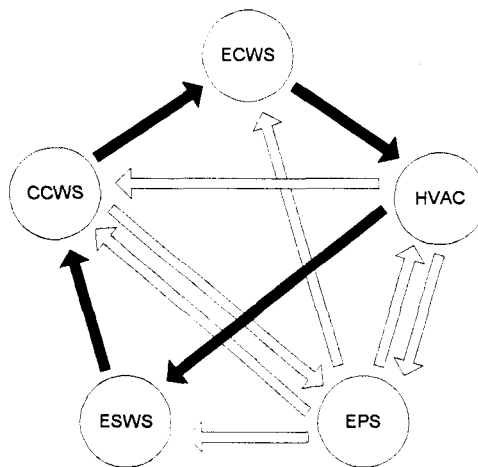
1단계 PSA의 최종 목적은 발전소 노심 손상 빈도 (Core Damage Frequency)를 계산하는 것이며 이를 위하여서는 모든 계통의 고장 수목이 함께 합쳐져 각 사고 경위에 대한 최소 단절군 (Minimal Cut Set)이 구하여 져야 한다. 이와 같이 고장 수목을 합치는 과정 중에 보조계통

(Support System) 간의 순환 논리 (Logical Loop)가 나타나게 된다. 이런 순환 논리는 최소 단절군을 구하기 위하여 반드시 해결 되어야만 한다.

기존의 PSA에서는 이를 위하여 순환 논리를 유발하는 부분을 갖지 않는 별도의 보조계통 고장 수목을 다시 작성하여 사용해 왔다 [1]. 그러나, 이 방법은 보조계통 간의 관계 검토, 이에 따른 별도 고장 수목의 재작성 등 추가적인 작업이 필요하며 또한 일부 최소 단절군이 생성되지 않는 약점을 갖고 있다. 이에 따라 한국 원자력 연구소 (KAERI)에서는 해석적으로 순환 논리를 푸는 방법을 개발하였으며 [2], 이를 KAERI에서 개발된 PSA용 코드인 KIRAP 코드에 구축하였다 [3]. 이에 따라 기존 방법의 약점을 극복하고 보조계통 고장 수목간의 순환 논리를 자동으로 풀 수 있게 되었다. 본 논문에서는 고장 수목간의 순환 논리를 풀기위하여 개발된 해석적 방법을 설명하며, 또한 이 방법을 실제 PSA에서 이용하며 나타난 몇가지 현상에 대하여 설명한다.

2. 해석적 방법을 이용한 순환 논리의 해결

가압 경수로 보조계통의 고장 수목간에 나타나는 전형적인 순환 논리가 그림 1에 나와 있다. 그림에서 검은 화살표로 표시된 바와 같이 ECWS는 HVAC를 보조하며, HVAC는 ESWS를, ESWS는 CCWS를, CCWS는 다시 ECWS를 보조하는 관계가 형성된다. 이와 같은 관계를 보조계통 간의 순환 논리라고 부른다.



ECWS: Essential Cooling Water System CCWS: Component Cooling Water System
 ESWS: Essential Service Water System EPS: Electrical Power Distribution System
 HVAC: Heating, Ventilation and Air Conditioning System

그림 1. 고장 수목간 순환 논리의 예

기존의 방법에서는 이들 관계 중 가장 상관 관계가 적은 부분을 찾아 이를 무시하는 방법을 사용하여 왔다. 예를 들어, 위 그림에서 HVAC과 ESWS의 상관관계가 가장 적다면, 이를 무시하고 HVAC과 관계가 없는 것으로 가정한 상태에서 ESWS의 고장 수목을 다시 작성하여 이를 이용하여 순환 논리를 풀어왔다. 이와 같은 방법은 앞에서 언급한 바와 같이 보조계통 간의 관계를 검토하여 가장 관련이 적은 부분을 찾아 내야 하며, 이에 따른 별도의 고장 수목을 작성하여야 하는 등 추가적 작업을 필요로 하였다. 또한, 일부 최소 단절군이 생성되지 않는 약점을 갖고 있다. 즉, 위의 예에 따르면 HVAC과 ESWS의 관계가 무시되었으므로, HVAC의 고장에 의하여 유발되는 ECWS의 고장을 나타내는 최소 단절군은 이 경우에 나타나지 않게 된다.

이와 같은 여러 문제점을 해결하기 위하여 KAERI에서 순환 논리를 푸는 해석적 방법을 개발했다. 해석적 방법에서는 고장 수목간의 관계를 일종의 경로 찾기 문제로 파악한다. 즉 고장 수목에서 기본 사건이나 게이트를 하나의 노드로 그들 사이의 연관 관계를 경로로 간주하는 것이다. 경로 찾기 문제에서 순환 논리란 어떤 노드에서 출발한 경로가 다시 자기 자신으로 돌아오는 경로를 의미한다. 개발된 방법에서는 각 노드로 부터 다른 노드로 갈 수 있는, 즉, 모든 경로를 찾아 그 중에 순환 논리를 유발하는 노드를 삭제하는 방법으로 순환 논리가 포함되지 않는 경로만을 찾아낸다. 순환 논리를 유발하는 노드는 다른 노드와의 중복임으로 그 물리적 의미가 없기 때문이다. KIRAP 코드에서는 어떤 정점사건 (Top Event)에 대하여 푼다고 할 때 그 정점사건을 하나의 출발 노드로 보고 이로부터 Top-Down/Depth-First 탐색 방법을 이용하여 모든 경로를 찾아내도록 하고 있다. 이와 같은 해석적 방법을 이용할 때 고장 수목이 어떻게 변하는 지가 다음의 그림 2와 3에 나와 있다.

그림 2는 서로가 서로를 보조하는 세 개의 보조 계통 A, B 및 C가 있다고 할 때, 계통 A에 대한 고장 수목이다. 그림에서 사각 노드는 각 보조계통의 정점 사건을, 원형 노드는 기본 사건이나 게이트를 의미하며 팔각형으로 표시된 노드는 순환 논리를 유발하는 부분을 표시한다. 즉, 그림의 예에서는 A, B 및 C 3개의 계통이 상호간에 순환 논리를 유발한다. 따라서 기존의 방법에서는 A, B 및 C 모든 계통에 대하여 별도의 고장 수목을 작성하여야 했다. 그러나 개발된 방법에서는 코드가 이와 같은 순환 논리를 유발하는 부분을 자동으로 점검하여 그 부분을 삭제한다.

그림 2와 같은 순환 논리를 갖는 고장 수목을 해석적 방법으로 풀면 그림 3과 같이 순환 논리를 갖지 않는 고장 수목이 나타나게 된다. 이 고장 수목을 풀음으로써 A 계통에 대한 최소 단절군을 구하게 된다.

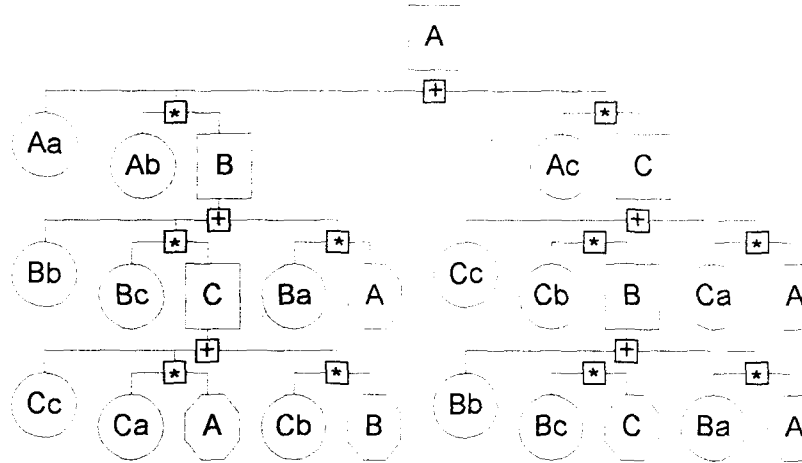


그림 2. 순환 논리를 갖는 고장 수목의 예

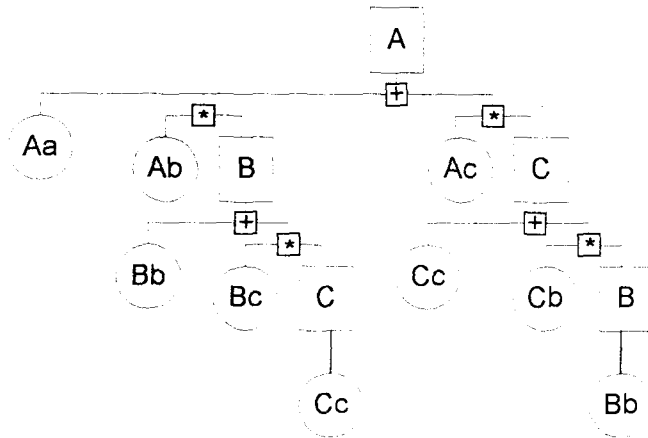


그림 3. 순환 논리가 삭제된 고장 수목의 예

3. 실제 PSA에의 적용

앞에서 설명한 해석적 방법은 가압 중수로인 월성 2/3/4호기 PSA에서 보조계통간의 순환 논리를 푸는 데 실제로 사용되었다. 먼저 월성 2/3/4호기 PSA에 사용된 사건 수목 및 고장 수목의 특징을 살펴 보면 다음과 같다.

- (1) 중수로의 계통수가 경수로에 비하여 많음에 따라 사건 수목의 수가 일반적인 경수로 PSA에 비하여 상당히 많다.
- (2) 중수로 PSA가 설계 PSA로 시작됨에 따라 그 신뢰도 자료가 매우 작은 부품단위까지 세분화 되어 있다. 따라서, 고장 수목의 크기가 경수로의 고장 수목에 비하여 훨씬 크다.

위와 같이 월성 2/3/4호기 PSA에 사용된 고장 수목의 크기가 크다고 하여도 논리적으로는 개발된 방법을 사용하는 데 문제가 없어야하나 실제 적용을 하는 데 있어서는 다음의 두가지 문제점이 발생하였다.

- (1) 해석적 방법을 사용할 때 고장 수목의 크기가 커짐에 따라 기존 방법에 비하여 계산 시간이 증가하였다.
- (2) 기존의 고장 수목에서는 시간에 대한 고려가 직접적으로 이루어 지지 않는다. 따라서 순환논리를 풀기 위한 새로운 고장 수목을 만들 때 이와 같은 시간에 대하여 고려되던 사항이 해석적 방법에서는 고려되지 않음으로써 비논리적 최소단절군이 생성되었다. 이와 같은 경우는 특히 전기관련 고장수목에서 발생한다. 예를 들어 모터구동밸브에 필요한 전원은 밸브를 열거나/닫는 순간에만 필요하다. 따라서 모터 구동 밸브 실패에 대한 최소 단절군에 전원의 Running Failure가 들어가는 것은 비논리적이다. 기존의 방법에서는 순환 논리 분석을 위하여 새로운 고장 수목을 작성할 때 이와 같은 부분을 미리 없앴으로써 비논리적 최소 단절군이 생성되지 않도록 하고 있다. 그러나, 해석적 방법에서는 이와 같은 수정 작업이 이루어 지지않으므로 비논리적 최소 단절군이 나타난다.

위의 두가지 문제점을 해결하기 위하여 다음과 같은 방법을 사용하였다.

- (1) 계산 시간의 감소를 위하여 중요한 몇 보조계통간의 순환 논리를 미리 풀어 그 최소 단절군을 File에 저장한 후 실제 사고 정량화 과정에서는 순환 논리를 푸는 대신 File에 저장된 최소 단절군을 이용하였다. 이에 따라 계산 시간 문제는 해결이 되었다.
- (2) 기존의 PSA에서는 전기관련 고장 수목이 단지 한가지 만이 존재하여 이를 Demand와 Running Component에 구분없이 사용하였다. 그러나 해석적 방법을 적용하기 위하여서는 각 경우에

대하여 별도의 고장 수목을 작성하여야 한다. 이는 추가적 작업으로 보일 수도 있으나, 고장 수목의 논리적 명확성을 확실하게 해준다는 점에서도 가치가 있다고 보여진다. 이와 같이 구축된 두 종류의 고장 수목을 Demand와 Running Component에 각기 사용함으로써 비논리적 최소 단절군은 나타나지 않게 된다. 이와 같은 고장 수목의 구분은 단지 전기관련 고장 수목만이 아니라 Demand와 Running Failure의 구분이 필요한 모든 계통에 필요하다.

4. 결 론

고장 수목간에 나타나는 순환 논리의 해결을 위한 해석적 방법을 개발하여 이를 KIRAP 코드에 구축하였다. 이는 기존의 PSA 코드가 순환 논리가 있을 경우 단지 이에 대한 정보만을 주며 이를 해결하지 못하던 것에서 진일보한 것으로 사고 경위 정량화에 소요되는 인력을 감소시키는 효과가 있었다. 기존의 고장 수목이 기존의 방법을 염두에 두고 만들었기 때문에 이들 고장 수목을 그대로 해석적 방법에 적용하면 상기한 바와 같은 문제점이 발생하였었다. 그러나, 앞에서 언급한 바와 같이 새로운 방법에 기초한 고장 수목은 해석적 방법을 적용하는데 문제가 없었다. 이 방법은 차후의 PSA에 많은 도움이 될 것으로 생각한다.

아울러, 이 방법은 현재 한국 원자력 연구소에서 추진되고 있는 PSA 자동화에도 중요한 역할을 할 것으로 생각된다.

5. 참고 문헌

- [1] G.A.Coles, T.B.Powers, "Breaking The Logic Loop To Complete The Probabilistic Risk Assessment," *Proceeding of PSA '89: International Topical Meeting Probability, Reliability, and Safety Assessment*, pp.1155-1160, Pennsylvania, U.S.A. 1989.
- [2] Yang, Joon-Eon et al., "An Analytic Method to Solve the Logical Loop between the Support Systems in the Fault Tree Model," *Proceeding of PSA '95: International Topical Meeting Probability, Reliability, and Safety Assessment*, pp.579-582, Seoul, Korea, 1995.
- [3] Han, Sang-Hoon et al., "PC-Workstation Based Level 1 PRA Code Package-KIRAP," *Reliability and System Engineering*, Vol.30, pp.313-322, 1990.