

## 화학공정에서의 안전성 자동 분석기 AHA의 개발에 관한 연구

강병관<sup>†</sup> · 강경욱\* · 이병우 · 서정철\*\* · 윤인섭

서울대학교 응용화학부

\*(주)쌍용정유

\*\*Soft DSC

(1998년 1월 5일 접수, 1999년 2월 8일 채택)

## A Study on the Development of AHA(Automatic Hazard Analyzer) for Chemical Processes

Byounggwan Kang<sup>†</sup>, Kyung Wook Kang\*, Byungwoo Lee,  
Jung Chul Suh\*\* and En Sup Yoon

Division of Chemical Engineering, Seoul National University, Seoul 151-742, Korea

\*SsangYong Oil Refining Co., Ltd., Ulsan 689-890, Korea

\*\*Soft DSC, Seoul 152-055, Korea

(Received 5 January 1998; accepted 8 February 1999)

### 요 약

화학공장은 수많은 장치들로 구성되어 있고 매우 복잡한 구조를 가지고 있다. 이러한 장치집약적인 특징으로 인해 화학공장은 항상 장치의 고장 또는 조업자의 실수로 인한 사고가 일어날 가능성을 안고 있다. 따라서 화학공정에서의 사고를 예방하고 안전을 확보하기 위해서는 잠재적인 사고 가능성 및 위험요인을 사전에 분석하고 예방하는 것이 중요하다. HAZOP 분석은 정성적인 평가 방법 중 가장 체계적이고 논리적인 방법으로 평가받고 있다. 그러나 이러한 안전성 평가를 위해서는 많은 전문가들로 구성된 전문인력, 자원, 시간이 필요하다. 따라서 인력과 시간을 줄이며 일관된 결과를 얻기 위해 위험성 평가의 자동화가 요구된다. 본 연구에서는 기존 방법론의 한계를 극복하기 위해서 화학공정의 안전성 분석 자동화 시스템을 구축하고자 한다. 이를 위해 일반적인 위험성 평가에 필요한 지식을 모델링한 다중모델 접근방법을 사용하여, 통상적인 물리적 위험성 이외에 반응성이나 인화성 등의 화학적 위험성까지 고려한 물질지식베이스, 공정을 연결관계에 따라 스트림(stream)별로 구분한 구조지식베이스, 하나의 장치를 기능과 변수를 구분하여 기능모델과 거동모델로 나누고 변수는 다시 위치에 따라 입력(inlet), 내부(internal), 출력(outlet)의 세 가지로 세분화한 장치지식베이스로 모델링하였고, 안전성 분석을 수행하는 세 가지의 추론 알고리즘-변수일탈을 중심으로 안전성 분석을 수행하는 변수일탈추론 알고리즘(Deviation Analysis Algorithm), 기능이상을 중심으로 안전성 분석을 수행하는 기능이상추론알고리즘(Malfunction Analysis Algorithm), 위 두 가지 분석결과를 바탕으로 공정의 물리적, 화학적 상태로부터 가능한 사고를 추론해 내는 사고분석알고리즘(Accident Analysis Algorithm)의 세 가지 알고리즘을 개발하여 화학공정의 안전성 분석 자동화 시스템 AHA(Automated Hazard Analyzer)를 개발하였다. AHA는 전문가 개발 도구인 G2를 이용하여 개발되었고, 제안된 시스템은 Olefin dimerization 공정의 도입부(feed section)에 적용되었으며 그 유용성이 확인되었다.

**Abstract** – There are many kinds of complicated equipment in chemical plants. So, the chemical plants always have high possibility of accidents. Hazard analysis is one of the basic tasks to ensure the safety of chemical plants, but it is an arduous, tedious, time-consuming work and requires multidisciplinary knowledge, and demands considerable cognitive load from the analysts. To overcome these problems, there have been various attempts to automate this work by utilizing computer technology, particularly knowledge-based technique. However, many of the past approaches are lacking in properties: safeguard consideration, accident diversity, cause and consequence diversity, pathways leading to accidents, and various hazard analysis reasoning. Therefore, in this study, deviation analysis algorithm, malfunction analysis algorithm, and accident analysis algorithm were proposed using multimodel approach of representing a chemical process adequately for hazard analysis in view of function, behavior, structure, and material property. Unit knowledge base is devised to model a process unit. It consists of unit behavior model and unit function model. In unit knowledge base, a process unit is modeled in different terms of variable and function. Variables describing unit are divided into inlet, internal, and outlet variables explicitly, by interconnecting behavior model and function model through these variables. This model represents physical hazards. In structure model, process is separated into several streams which represent different connections. In material knowledge base, chemical properties are considered according to NFPA code. A hazard analysis system, AHA, was developed on G2. The system was applied to feed section

<sup>†</sup>E-mail : chempire@pslab.snu.ac.kr

of olefin dimerization. And the case study was solved with AHA system successfully.

Key words : Automated Hazard Analysis, Multimodel, Deviation Analysis, Malfunction Analysis, Accident Analysis

## 1. 서 론

화학공장에는 수많은 장치들이 있고 매우 복잡한 구조를 가지고 있다. 이러한 화학공장은 장치집약적인 특징으로 인해 항상 장치의 고장 또는 조업자의 실수로 인한 사고가 일어날 가능성을 안고 있다. 또한 화학공장은 여러 가지 위험한 물리적, 화학적 특성을 가진 물질을 다루고 운전상태가 고온, 고압일 가능성이 크므로 이러한 곳에서의 사고는 개인적인 상해뿐 아니라 화재, 폭발, 유독 물질 누출 등 대형사고로 일어나기가 쉽고, 이는 공장 내 지역뿐만 아니라 주변지역과 지역주민에 대해서도 직접, 간접적인 피해를 입히게 된다. 따라서 화학공정에서의 사고를 예방하고 안전을 확보하기 위해서는 잠재적인 사고 가능성 및 위험요인을 사전에 분석하고 예방하는 것이 중요하다. 공장에서 안전성을 확보하기 위한 노력은 설계 단계에서부터 시작하여 조업 단계, 공정의 변경 단계 등 공정의 전체 수명주기동안 진행되고 있다[1-4, 8-11, 13, 14].

이러한 안전을 위한 활동들은 크게 온라인 이상 진단(on-line fault diagnosis)과 오프라인 위험성 평가(off-line hazard analysis)로 나뉘어진다. 온라인 이상 진단은 공정에서 발생한 이상이 사고로 발전하는 것을 예방하는 것이 목적이고 오프라인 위험성 평가는 공장의 설계 단계나 기존 공정에 변화가 있는 경우에 수행하게 된다. 위험성 평가는 이상 진단에 비해 이상의 원인(fault, symptom, deviation)뿐 아니라 결과(consequence)까지 알아낼 수 있고, 또한 본질적으로 안전한 공장을 확보하기 위한 설계의 변경사항 등을 알 수 있기 때문에 매우 중요한 작업이다.

지금까지 화학공장의 잠재 위험요소를 알아내기 위한 다양한 위험성분석기법들이 고안되어 사용되어져 왔다. 대표적인 것으로는 Hazard and Operability Study(HAZOP), Failure Mode and Effects Analysis(FMEA), What-if Analysis, Checklists, Preliminary Hazard Analysis(PHA) 등이 있다. 이중에서도 HAZOP 분석은 정성적인 평가 방법 중 가장 체계적이고 논리적인 방법으로 평가받고 있다. 이러한 HAZOP 분석과 같은 안전성 평가를 위해서는 많은 인력, 자원, 시간이 필요하다. 따라서 전문가의 인력과 시간을 줄이며 일관된 결과를 얻기 위해서는 위험성 평가의 자동화가 절실히 요구된다.

기존에 제시된 위험성 평가 자동화 시스템들은 안전성 분석을 위한 모델링의 한계로 인해 공정에 장치되어 있는 안전 장치들을 고려하지 못하거나 고려하기 어렵게 되어 있었다. 그리고 화학공장의 위험요소는 고온, 고압 같은 물리적 위험 요소(physical hazards)들뿐만 아니라 물질의 독성, 폭발성, 인화성 등의 화학적 위험 요소(chemical hazards)로 분류할 수 있는데, 지금까지 제안된 시스템들은 대개 물리적 위험 요소를 알아내는데 국한되어 있었다. 또한 공정의 잠재 사고를 관리하기 위해서는 단순히 사고의 원인과 결과를 알아내는 것보다는 근본 기능이상에서부터 실제 사고에 이르는 모든 경로를 알아내야 체계적인 대비가 가능하다. 그러나 기존의 시스템들은 단순히 변수 이탈의 원인과 결과만을 제시해 주기 때문에 사고의 중간 과정을 알기 어렵다. 또한 기존의 시스템들은 자동화의 대상이 HAZOP 방법론에 국한되어 있고 주어진 변수 이탈에서 추론을 시작하거나, 이상(fault)을 가정하여 위험 요소를 알아낸다. 이 두 가지를 서로 보완하면 좀 더 나은 안전성 분석 결과를 얻게 될 것이다.

본 연구에서는 이러한 기존 방법론의 한계를 극복하기 위해서 일반적인 위험성 평가에 필요한 지식을 모델링한 다중모델 접근방법을 사용하였으며, 사고의 메커니즘을 분석하고, 이를 바탕으로 안전성 분

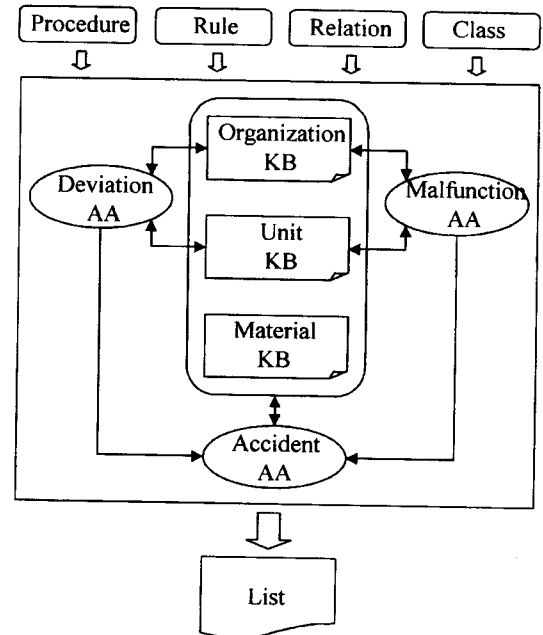


Fig. 1. Architecture of AHA.

석을 수행하는 추론 알고리즘을 개발하여 화학공정의 안전성 분석 자동화 시스템을 구축하고자 한다.

## 2. 안전성 자동분석기 AHA(Automatic Hazard Analyzer)

### 2-1. AHA의 구조

본 연구에서 개발한 안전성 자동분석시스템 AHA(Automatic Hazard Analyzer)의 개념적인 구조는 Fig. 1과 같다.

본 시스템은 전체 Knowledge Base와 Analysis Algorithm을 총괄하고 제어하는 Inference Engine과 세 개의 Knowledge Base, 그리고 세 개의 Analysis Algorithm으로 이루어져 있다. Deviation Analysis Algorithm과 Malfunction Analysis Algorithm을 추론할 때는 Organizational Knowledge Base와 Unit Knowledge Base를 사용하고, Accident Analysis Algorithm은 Organizational Knowledge Base와 Material Knowledge Base를 사용하여 추론한다.

### 2-2. AHA의 특징

본 시스템은 Ram 64 Mb의 IBM PC에서 전문가 시스템 개발 도구인 G2를 사용하여 구현하였다. 모델링에 사용된 장치와 개체들은 모두 class를 사용하여 정의하였고 상위 class와 하위 class를 정의함으로써 공통적인 속성을 상속받을 수 있게 하였다[7]. 각각의 Knowledge Base, Analysis Algorithm들과 Inference Engine에 대해서는 각각의 workspace를 부여하였고 장치마다 workspace를 부여함으로써 새로운 장치의 추가와 수정이 용이하도록 하였다. 또 Organizational Knowledge Base에서의 각각의 unit의 연결관계는 G2의 특성을 이용하여 연결과 변형, 단절이 용이하며 연결된 정보도 자동으로 기억하게 된다. Table 1에 각 Knowledge Base들과 Analysis Algorithm들의 표현

Table 1. Knowledge representation methods

Knowledge Base & Analysis Algorithm	Models	Representation
Unit Knowledge Base	Unit Behavior Model	Class & Relation
	Unit Functional Model	Class & Relation
Material Knowledge Base	Hazard Index	Class & Relation
Organization Knowledge Base		Frame
Accident Analysis Algorithm		Rule & Relation
Malfunction Analysis Algorithm		Rule & Procedure
Deviation Analysis Algorithm		Rule & Procedure

방법을 요약하였다.

Unit Behavior Model과 Unit Functional Model은 자료구조 중의 tree구조를 갖게 되므로 이를 이용하여 부모와 자식에 해당하는 연결 관계와 정보를 relation으로 표현하였다. Tree 구조는 한 노드가 여러 개의 하위노드를 가지는 구조를 효과적으로 탐색하기 위한 자료구조로서 G2에서는 relation을 통하여 상위노드와 하위노드의 모든 연결 정보를 기억할 수 있게 되며 탐색시에 이를 이용하게 된다. 즉 하나의 상위노드에 이르면 이 노드에 연결된 하위노드 모두를 찾아낼 수 있고 그 하위노드들에 각각 연결된 하위노드를 또한 모두 찾아낸다. 반대로 하위노드에서 상위노드로 탐색시에도 하나의 하위노드에 연결된 상위노드를 relation에 의한 정보를 가지고 찾게 되며 계속해서 그 위의 상위노드도 찾는다.

Unit Behavior Model과 Unit Functional Model에서의 variable의 표현은 일정한 pattern을 갖도록 하여 탐색을 용이하도록 하였다. 또한 workspace를 다르게 하였기 때문에 같은 장치에 대해서는 같은 variable 이면서도 동시에 두 개의 workspace에 존재해야 한다. 그리고 모델화 하기 위해 장치마다 새로운 workspace를 정의하였으므로 장치의 숫자만큼 variable 수가 늘어난다. 그러므로 이름을 달리하면서도 구별이 쉽고 탐색의 편의를 위해 예를 들어 F-PIPE-IN-P-HIGH와 B-PIPE-IN-P-HIGH 같이 표현하였다. 하이픈 '-'은 탐색시 각 단어사이를 구분하는 역할을 하게 되고 각 단어는 다음과 같은 의미를 가진다.

F : Function model에서의 variable

B : Behavior model에서의 variable

PIPE : 장치의 이름

IN : Inlet variable

P : Pressure

HIGH : High deviation

Analysis algorithm 수행시에는 이들 variable의 이름을 '-'을 이용하여 단어별로 분류하여 각 장치마다 일어나는 deviation을 구분하게 되고 deviation의 전달을 하게 될 때는 'deviated'라는 attribute를 주어서 deviation이 발생하면 deviated가 'on'이 되고 tree구조를 이용하여 다음 노드에 deviated를 on 시킨 후 자신의 deviated를 'off'로 만들면서 이상을 전달한다.

장치의 unit function model과 unit behavior model은 각 장치마다 한 개씩인데 전체 공정도에서는 여러번에 걸쳐 같은 장치가 반복된다. 또한 장치와 variable의 이상의 전달이 독립적이기 때문에 장치가 나오는 순서를 구별하면서도 같은 model을 이용하고 또한 서로 보조를 맞춰주는 것이 필요하다. 그래서 장치에는 PIPE-1과 같이 나오는 순서에 따라 번호를 덧붙여 구별한 후 장치이름에서 번호를 제외한 이름과 variable의 장치이름을 비교하여 이름이 서로 같을 때만 이상을 전달하게 함으로써 같은 장치가 나중에 또 나오더라도 혼란이 없게 하였다.

주요한 workspace는 다음과 같다.

- Accident Analysis Algorithm
- Malfunction Analysis Algorithm
- Deviation Analysis Algorithm
- Class
- Main Inference
- Unit Knowledge Base
- Organizational Knowledge Base
- Material Knowledge Base

AHA에서는 이러한 workspace들을 hierarchical하게 표현하여 공정의 전체구조를 알기 쉽게끔 하였다.

### 3. 다중모델접근법을 이용한 AHA의 지식모델링

공정을 모델링하는데에는 여러 가지 방법이 있으나, 위험성 평가를 하기 위해서는 공정에 존재하는 위험요소와 사고요소들에 초점을 맞추어 모델링하여야 할 것이다. 이러한 요소들은 매우 다양하며 이들을 한 가지 방식의 모델만을 이용하여 모델링하기는 매우 힘들다. 이러한 이유로 위험성 평가에 적합한 모델링으로서 다중모델접근법을 사용하였다. 이는 공정모델을 구조지식베이스, 물질지식베이스, 장치지식베이스의 서로 다른 세 측면으로 나누어 독자적으로 모델링하였음을 의미한다[4]. 이러한 지식베이스를 구축하기 위한 지식은 대개 다음의 세 가지로부터 만들어진다고 가정하였다.

- 조업 전문가의 과거 경험(prior experience of experts)
- 공정의 과거 조업 데이터(process operation history)
- 기본적인 물리화학적 원리(fundamental physical principles)

#### 3-1. 구조지식베이스(Organizational Knowledge Base)

구조지식베이스(organizational knowledge base)는 단위 장치들의 공간적인 배치나 연결관계에 관한 지식을 가지고 있다. 이 지식은 하나의 공정이 어떤 단위 장치들로 구성되어 있고, 그들이 서로 어떻게 연결되어 있는지를 나타낸다.

이러한 정보는 P&ID같은 공정 도면과 사용자료부터 제공되어 완성된다. 이는 각각의 장치마다 장치의 종류명, 그 장치의 이름, 추론 알고리즘에 의해 이상이 전파되는 경로의 정보를 전달하기 위한 속성, 전후 장치의 연결정보 등을 가지게 된다. 각 장치들은 종류에 따른 클래스를 정의한 뒤 개체를 만들어 구현하게 된다. Main stream과 side stream, control loop는 하나의 workspace에 표현되고 클래스의 종류와 색깔로 구별하도록 하였다.

구조지식베이스는 추론알고리즘에서 이상의 전파과정을 나타내고 사고발생장치를 파악할 수 있게 해주며 장치마다 장치의 특성에 따른 사고의 다양성을 표현할 수 있다.

Fig. 2는 사례연구에서 다루게 될 feed section of olefin dimerization의 구조지식베이스이다. 그림에서 인터페이스는 공정도면과 흡사하며 각 장치를 클릭하여 해당 장치의 정보를 볼 수 있고 장치의 연결상태를 한 눈에 볼 수 있음을 알 수 있다.

#### 3-2. 물질지식베이스(Material Knowledge Base)

본 연구에서는 NFPA(National Fire Protection Association) code [5, 6]를 이용하여 물질지식베이스를 구현하였다.

물질지식베이스는 위험성 지수(hazard indices)를 나타내는데 위험성 지수는 NFPA code에 따라서 각 물질에 대한 인화성 지수(flammability hazard rating(Nf)), 유해성 지수[Health hazard rating(Nh)], 반응성

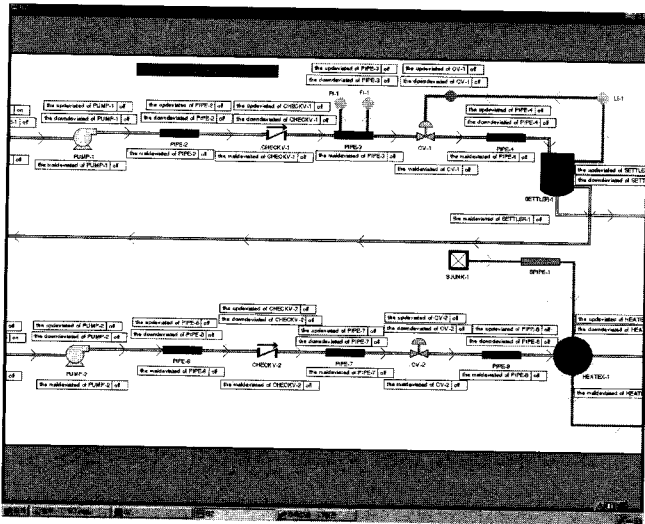


Fig. 2. Organizational knowledge base.

지수[Reactivity hazard rating(Nr)]로 되어 있다. 각각의 지수는 0부터 4까지의 크기를 가지며 지수가 커짐에 따라 위험 정도가 높아짐을 나타낸다.

물질을 대표하는 하나의 클래스를 정의하여 각각의 물질마다 하나씩 개체를 만들어 위의 3가지 지수를 부여하였고 장치에 독립적으로 하였다. 그리고 난 후 각각의 장치를 지나게 되는 물질의 종류를 장치마다 따로 초기화하였기 때문에 장치를 지나서 물질이 바뀌면 초기화를 다시 해주면 되고 물질이 추가되거나 변동되기 때문에 따르는 물질지식베이스 자체의 변동 등과 같은 부가적인 작업은 없고, 다만 기존에 구축되지 않은 물질만 추가해서 넣어주면 된다.

물질지식베이스는 추론알고리즘에 의해 찾아진 사고에 대해 부가적인 정보를 제공함으로써 1차적으로 일어난 사고가 물질의 특성에 따라 부가적인 사고로 연결될 가능성에 대한 기능을 하게 된다.

물질지식베이스를 구현한 모습이 Fig. 3에 나타나 있다. 그림에서 이 지식베이스가 각 물질에 대해서 인화성, 유해성, 반응성 지수의 정보

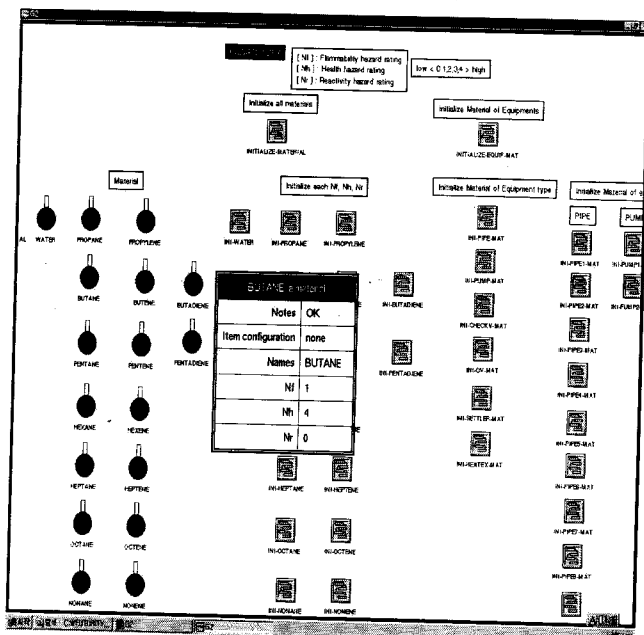


Fig. 3. Material knowledge base.

를 담고 있음을 알 수 있다.

### 3-3. 장치지식베이스(Unit Knowledge Base)

물리적 위험 상황을 표현하는 모델을 만들기 위해서 먼저 공정에서 위험한 상황이 전개되는 순서를 분석해 보면 다음과 같이 순서를 정할 수 있다.

- (1) 한 공정 장치(basic unit)에서 기능 이상이 발생한다.
- (2) 기능 이상이 그 장치의 상태변수에 일탈을 유발한다.
- (3) 상태변수의 일탈이 upstream과 downstream의 장치의 변수에 일탈을 야기한다.
- (4) 장치와 upstream, downstream의 기능 이상과 변수 일탈이 사고(accidents)와 연결된다.

이렇게 사고발생순서를 정하게 되면, 이를 표현하기 위해서 다음의 네 가지 종류의 지식이 필요하게 된다.

- (1) 단위 장치에서 기능 이상과 변수와의 관계
- (2) 단위 요소 변수들의 인과 관계
- (3) 단위 장치들의 전체 공정에서의 서로 연결 상태
- (4) 기능 이상, 변수 일탈과 사고의 관계

이를 위하여 장치 거동 모델(unit behavior model)과 장치 기능 모델(unit function model)로 구성되는 장치지식베이스를 구축하였는데 이들은 각각 위의 (1)과 (2)에 해당하는 지식을 표현한다. 참고로 (3)의 지식은 앞에서 설명한 구조지식베이스에서, (4)의 지식은 뒤에서 설명할 사고분석알고리즘 내에서 (1)에서 (3)까지의 지식베이스를 모두 활용하여 결정한다.

#### 3-3-1. 장치거동모델(Unit Behavior Model)

위험성 평가에서 필요한 거동모델은 각 단위 요소 전후, 변수들의 이상 전과 관계(인과 영향 관계, causal influence)를 표현하는 것이다. 공정의 상태를 나타내는 여러 변수 중에서 위험성 평가에서 관심있는 변수인 압력[pressure(P)], 온도[temperature(T)], 유량[flow rate(F)], 조성[composition(C)], 액위[level(L)], 반응 변수[reaction(r)] 등으로 이상 전과 관계를 표현한다. 각각의 변수는 high와 low로 구분되고 위치에 따라서도 inlet, internal, outlet으로 세분화되어 있다. 변수에 해당하는 클래스를 먼저 정의한 후 각 변수의 위치와 종류는 이름으로 구별하였다. 이들간의 관계는 relation으로 초기화하였다.

파이프에 대한 장치거동모델을 구현한 모습이 Fig. 4에 나타나 있다. 각 변수의 위치(inlet, internal, outlet)에 따라 파이프의 각 변수 거동의 원인과 결과 관계가 화살표로 표시되어 있음을 알 수 있다.

#### 3-3-2. 장치기능모델(Unit Function Model)

장치기능모델은 단위 장치가 그 장치 안에 존재하는 변수에 대해 어떤 기능을 하는가에 대한 정보를 나타낸 모델이다[12].

이 모델은 의미있는 inlet, internal, outlet 변수별로 이들이 high malfunction과 low malfunction 상태에 이르게 하는 원인들을 결정하고 이 계층관계(hierarchy)를 표시한다.

같은 기능 이상이라도 여러 변수에 영향을 줄 수 있고 장치의 종류와 전문가들의 전문성, 분석의 정도에 따라 기능이상( malfunction)의 가짓수와 깊이는 달라지게 된다.

우선 변수를 나타내는 변수와 기능을 나타내는 변수를 다른 클래스로 정의하였고 여기서의 변수는 위의 장치거동모델의 변수와 같은 클래스이나 구별을 위해 다른 이름을 붙이게 된다. 그리고 변수와 기능과의 관계는 relation으로 구현하였다. Fig. 5에 구현된 장치기능모델의 모습이 있다. 이 모델에서는 파이프의 inlet, internal, outlet 변수들의 변이에 대해 이 변이들을 일으킬 수 있는 기능이상(malfunction)들을 직접적인 것들(immediate malfunction)과 2차적인(intermediate

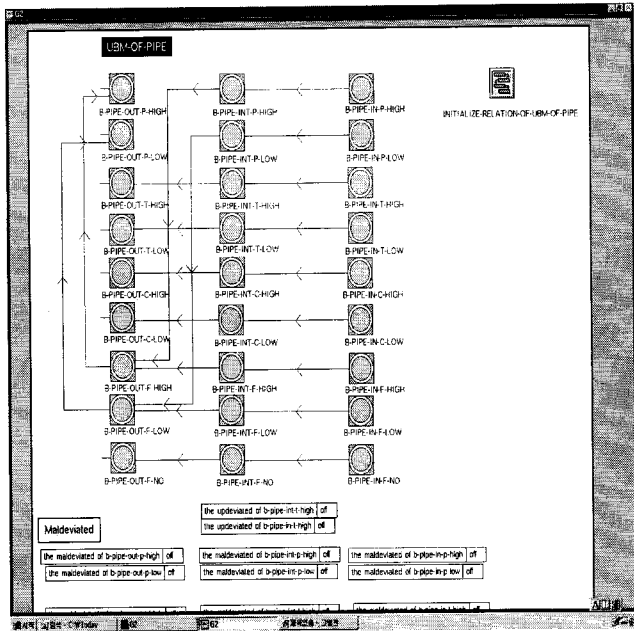


Fig. 4. Unit behavior model.

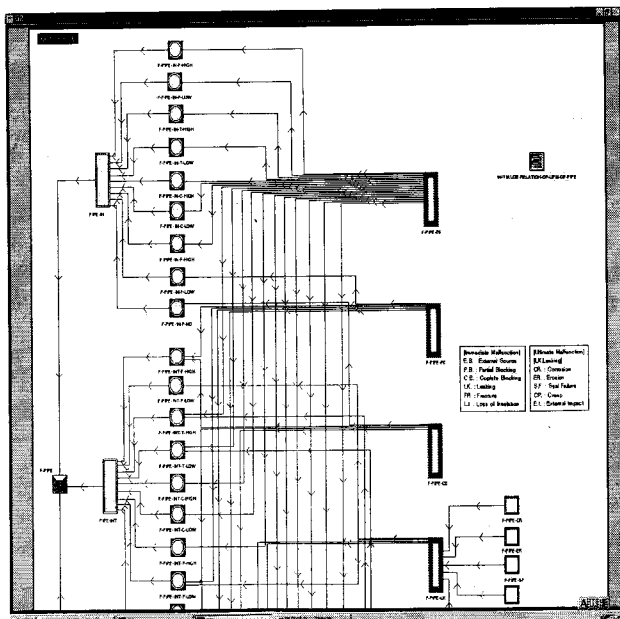


Fig. 5. Unit function model.

malfunction) 혹은 근원적인 것들(ultimate malfunction)로 나누어 표현하였음을 그림에서 볼 수 있다.

#### 4. AHA의 추론 알고리즘

지금까지 소개한 AHA를 구성하는 지식베이스는 그 자체의 구성도 중요하지만, 이 지식베이스들을 어떻게 활용하여 사고의 발생과정과 결과를 추론해내느냐 하는 추론알고리즘의 개발은 훨씬 더 중요한 의미를 갖는다. 본 논문에서는 AHA의 추론알고리즘으로서 변수일탈로부터의 사고추론알고리즘(deviation analysis algorithm), 장치기능이상으로부터의 사고추론알고리즘(malfunction analysis algorithm), 또한

이 두 가지 알고리즘의 분석결과와 실제 사고의 형태를 연관지어주는 사고분석알고리즘(accident analysis algorithm)의 세 가지 알고리즘을 개발하였다.

##### 4-1. Deviation Analysis Algorithm

Deviation analysis algorithm은 변수 일탈(given deviation)로부터 추론이 시작되어 그 단위 장치의 기능 모델(unit function model)에서 주어진 변수 일탈의 원인이 되는 기능 이상(root malfunction)을 찾는다. Root malfunction은 없을 수도 있고, 여러 개일 수도 있다. 다음으로 장치 거동 모델(unit behavior model)로부터 주어진 deviation의 원인이 되는 다른 변수들의 일탈 상태를 모두 찾는다. 이것을 cause deviation이라고 부른다. 이 과정은 다시 inlet variable, internal variable, outlet variable의 영향을 받는 경우로 나누어 구조 지식 베이스(organizational knowledge base)와 장치의 거동 모델(unit behavior model)로부터 cause variable deviation을 알아내고, 또한 각 unit의 cause deviation이 결정되면 각 장치의 기능 모델에서 그 장치에서 확인된 장치의 cause deviation에 대한 기능 이상(cause malfunction)을 알아낸다. 이렇게 모든 deviation과 malfunction을 알아내고 나면 accident analysis algorithm에 의해 가능한 사고를 추론하며 그에 관한 safeguards가 제시된다.

Internal variable이 관련된 경우에는 그 unit의 behavior model에서 다시 cause variable deviation을 찾는다. 확인된 cause deviation은 다시 inlet variable, internal variable, outlet variable이 될 수도 있다. Inlet이나 outlet variable의 경우 앞에서 설명한 절차에 따라 다음 추론을 진행하고, 다른 internal variable의 경우에는 다시 이 과정을 반복하게 된다.

위의 추론 과정을 거쳐 주어진 변수 일탈에 대한 원인 규명이 끝나면, 주어진 변수 일탈이 다음 장치에 미치는 파급 효과(propagation effect)를 알아내기 위해 다음 단위장치의 거동 모델에 주어진 변수 일탈을 cause deviation으로 하여 그 effect deviation을 알아낸다. 이 과정을 계속 후속 단위 장치에 적용하여 분석자가 분석을 원하는 장치까지 파급 영향을 알아낼 수 있다.

다음으로 deviation analysis algorithm은 accident analysis algorithm을 부른다. 구체적인 추론과정이 Fig. 6에 있다.

##### 4-2. Malfunction Analysis Algorithm

Malfunction analysis algorithm은 사용자가 지정한 장치의 기능 이상으로부터 추론을 시작한다. 장치의 기능 모델로부터 주어진 이상 기능이 발생시키는 변수에 대한 영향을 알아낸다. Given malfunction에 의해 발생하는 variable deviation은 한 가지일 수도 있고, 여러 개일 수도 있다. 다음으로 이 변수 일탈이 주변 장치들로 영향을 미치는 것을 구조 지식 베이스와 장치 거동 모델로부터 알아낸다. Deviation analysis와는 달리 주어진 malfunction 이외의 다른 장치에서의 이상 기능은 고려하지 않는다. 주어진 기능 이상에 의해서 영향을 받는 모든 단위 장치의 물리적 상태가 결정되면 accident analysis algorithm을 부른다.

Malfunction analysis algorithm을 Fig. 7에 나타내었다.

##### 4-3. Accident Analysis Algorithm

Accident analysis algorithm은 모든 장치의 물리적 상태와 물질에 대한 정보로부터 발생 가능한 사고들을 추론하는 것이다. 이를 위해서 material knowledge base로부터 공정 내 장치들이 지니는 물질에 대한 인화성 지수, 반응성 지수, 유해성 지수에 관한 정보를 얻고, 확인된 기능 이상과 변수 일탈로부터 관련있는 위험 요소를 알아낸다. Accident analysis algorithm에는 모든 생각 가능한 형태의 사고가 포

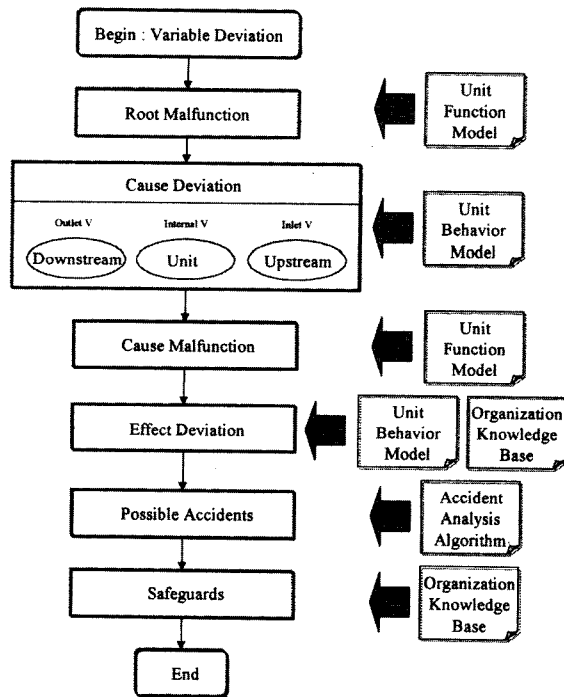


Fig. 6. Deviation analysis algorithm.

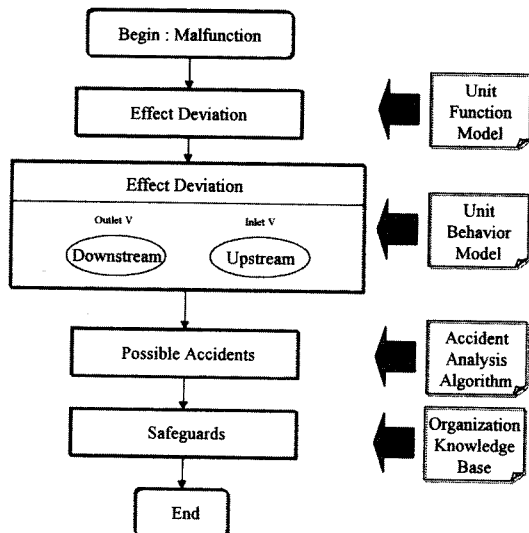


Fig. 7. Malfunction analysis algorithm.

현될 수 있다. 그 중에서 전형적인, 세 가지 accident type은 malfunction과 위험물질에 의해 일어나는 사고, 변수이탈과 위험물질에 의해 일어나는 사고, 변수이탈과 장치의 특성으로 일어나는 사고 등이다.

Accident analysis algorithm은 공정 장치의 malfunction, deviation, characteristic, material property를 연계하여 보다 다양한 형태의 사고를 추론해낼 수 있다. Fig. 8에 accident analysis algorithm을 나타내었다.

이 추론과정까지 끝내면 마지막으로 전체 추론 엔진에서는 추론된 위험 상황에 대한 안전 장치(safeguards)가 존재하는가를 알아내기 위해 구조 지식 베이스로부터 안전 장치의 배치 상태에 대한 정보를 얻고, 각 사고상황에 대하여 존재하거나 존재해야 할 안전장치의 종류를 앞의 모든 분석과정과 함께 분석결과로서 출력하게 된다.

화학공학 제37권 제3호 1999년 6월

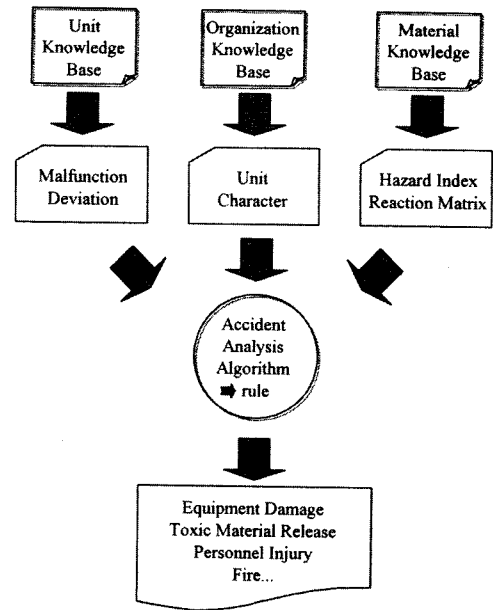


Fig. 8. Accident analysis algorithm.

## 5. 사례 연구

사례연구로서 olefin dimerization plant의 도입부 공정의 위험성 평가를 실시하였다. 본 공정은 F.P. Lee의 "Loss Prevention in the Process Industries"에서 인용하였다[8]. 이 공정은 Venkat가 제안한 HAZOPEXPERT에서 예제로 사용되기도 하였다[14]. 이 공정에서는 약간의 물이 섞여있는 알켄과 알칸의 혼합물이 중간 저장 탱크로부터 buffer/settling 탱크로 연속적으로 운송된다. 이 혼합물에 섞여 있는 물은 dimerization reaction에 나쁜 영향을 미치기 때문에 feed/product 열 교환기를 거치기 전에 settling tank에서 수작업으로 일정한 간격을 두고 제거한다. Fig. 9에 본 공정을 나타내었다.

Table 2는 이 공정의 feed section의 settling tank 직전부분의 pipe에서 'no flow'라는 deviation이 일어났을 경우에 그 variable의 deviation의 전파과정을 보여주는 결과의 일부이다. 연구결과는 어떤 한 unit에서 변수가 이탈이 일어났을 경우에 이 이탈이 다른 unit으로 어떻게 전파되는가 하는 경로를 보여줄 수 있다. 이것은 분명히 이전의 HAZOP이나 HAZOPEXPERT에서 나온 결과보다 훨씬 명확한 결과를 나타내 줄 수 있다. No flow를 입력으로 Deviation analysis를 수행하면 먼저 파이프의 기능 모델에서 이들의 root malfunction을 찾는다. 그 결과 CB(complete block), FR(fracture)가 제시된다. 다시 알고리즘은 파이프의 거동 모델로부터 cause deviation으로 no inlet flow를 인식한다. 그 다음 inlet variable의 영향을 받으므로 구조 지식 베이스로부터 앞의 장치 control valve를 인식한다. 이 밸브의 거동 모델에 no outlet flow를 effect deviation으로 입력하여 cause deviation으로 no internal flow를 찾고, 기능 모델로부터 이에 대한 cause malfunction을 알아낸다. 이러한 과정을 반복하여 주어진 변수 이탈의 원인들을 모두 알아내게 된다.

주어진 변수 이탈의 파급 영향을 알아보기 위해서 알고리즘은 파이프의 거동 모델에 cause deviation으로 no internal flow를 입력하여 no outlet flow를 인식한다. Outlet variable에 영향을 미치므로 구조 지식 베이스로부터 다음에 연결된 settling tank를 인식하고 이 장치의 거동 모델에 no inlet flow를 cause deviation으로 하여 effect deviation인 no internal flow를 알아낸다. 이러한 과정을 반복하여 모든 장

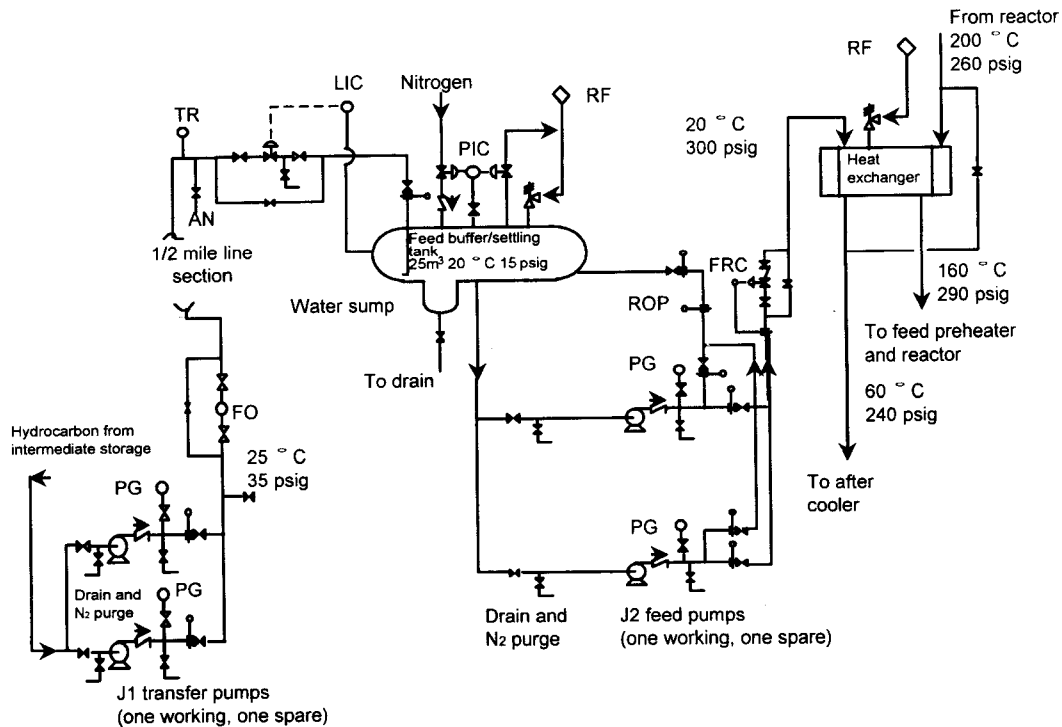


Fig. 9. Feed section of olefin dimerization plant.

Table 2. Conventional HAZOP and HAZOPEXPERT's result

Process variable deviation: No flow in pipe-4	
Conventional HAZOP Study Causes:	
No hydrocarbon available at intermediate storage	
Pump-1 fails	
Line blockage or LCV fails shut	
Line fracture	
HAZOPEXPERT's Causes:	
No flow into the 'hydrocarbon-from-storage' HAZOP connection post from upstream units	
Cavitation or failure of pump-1	
Pipe blockage, LCV-1 or NRV-1 fails closed	
Pipe fracture	
Conventional HAZOP Study Consequences:	
Loss of feed of reaction section and reduced output	
Polymer formed in heat exchanger under no flow conditions	
Pump-1 overheats	
Hydrocarbon discharged out	
Pump-2 overheats	
HAZOPEXPERT's Consequences:	
No flow to the downstream units from the 'feed-to-preheater-and-reactor' HAZOP connection post	
High temperature in the tube side of the heat exchanger	
The 'dimer-product' polymerizes to undesired polymer	
Pump-1 overheats	
Pipeline subjected to surge pressure	
Fire hazard due to the release of flammable process materials alkane and alkene	
Pump-2 overheats	
High pressure in the tube side of 'feed-product-heat-exchanger' and in pipe-12 due to higher temperature, since volatile process materials alkane, alkene and dimer-product are present	
Fire hazard in the tube side of heat exchanger and pipe-12 due to high temperature and the presence of the flammable process materials alkane, alkene and dimer-product	

치에서의 effect deviation을 인식하게 된다.

위의 분석 결과 공정의 모든 장치에서 물리적인 상태가 결정되고, 알고리즘은 accident analysis를 수행한다. Accident analysis에서는 pressure buildup, material release, fire, running dry 등의 사고로 추론된다. Settling tank의 running dry에 대해 level controller나 level indicator가 안전 장치로 제시된다.

Table 2에서는 문헌에서 소개된 기존의 HAZOP 및 HAZOPEXPERT의 실행결과 비교를 나타내었고[13], Table 3에서는 이 부분에 해당하는 AHA의 수행결과를 나타내었다. 또한 Fig. 10-12에서는 세 가지 분석결과를 각각 트리 형식으로 나타낸 그림을 보여주고 있다. 본 AHA 시스템의 결과는 이전의 다른 시스템의 결과와는 달리 사고의 시작(근본 기능 이상)에서부터 실제 사고에 이르는 경로를 모두 제시하여 주고 기존의 설치되어 있는 안전 장치가 적절한지를 검증해주고 있다. 또한 중간 추론 과정을 제시함으로써 추론에 대한 신뢰성을 높여준다.

그림에서 볼 수 있듯이 AHA의 위험성분석 결과가 보여주는 가장 큰 특징은 사고가 전개되는 경로가 나타나고 그 가운데 적절한 안전 장치가 있는지를 확인할 수 있게 하여 설계 또는 조업변경 단계에서 보다 안전한 공장을 만들기 위한 정보를 제공할 수 있게 한다는 점이다. 전통적인 HAZOP의 위험성분석방법은 사고의 발생경로보다는 사고의 원인과 결과에 초점을 맞추기 때문에 사고가 발생하는 구체적인 경로가 결과로서 제시되지 않는다. 이것은 HAZOP기법을 자동화한 HAZOPEXPERT에서도 마찬가지로 드러난다. 그러나 AHA 시스템의 위험성분석은 장치기능이상 일어났을 때 각 공정장치를 거치면서 발생하는 모든 장치이상 및 변수변이가 모두 결과로서 제시되므로 시스템을 사용하는 사용자가 사고에 대해 보다 잘 이해할 수 있다. 또한 AHA는 기존의 시스템에서 제공하지 않던, 사고예방을 위한 적절한 안전장치를 결과로서 제공하기 때문에, 단순히 일어날 수 있는 사고의 원인과 결과만 제공하는 기존의 위험성평가에 비해 진일보한 장점을 가지고 있다.

전문가들이 분석을 수행하는 경우에는 시간이 오래 걸리고, 인간의 특성상 주의력이 약해지는 등의 문제 때문에 분석 결과에 신뢰가 떨어

Table 3. Part of AHA's results of the faults of olefin dimerization plant

<p>deviation-up-list  B-PIPE-INT-F-NO is the effect of B-PIPE-INT-F-NO in PIPE-4  B-CV-OUT-F-NO is the effect of B-CV-INT-F-NO in CV-1  B-CV-INT-F-NO is the effect of B-CV-IN-F-NO in CV-1</p> <p>deviation-up-fd-list  B-PIPE-INT-F-NO is equal to F-PIPE-INT-F-NO in PIPE-4  B-PIPE-INT-F-NO is equal to F-PIPE-INT-F-NO in PIPE-4  B-CV-OUT-F-NO is equal to F-CV-OUT-F-NO in CV-1  B-CV-INT-F-NO is equal to F-CV-INT-F-NO in CV-1  B-CV-IN-F-NO is equal to F-CV-IN-F-NO in CV-1</p> <p>deviation-up-imal-list  F-PIPE-INT-F-NO is the deviation variable of F-PIPE-CB in PIPE-4  F-PIPE-INT-F-NO is the deviation variable of F-PIPE-FR in PIPE-4  F-PIPE-INT-F-NO is the deviation variable of F-PIPE-CB in PIPE-4  B-CV-OUT-F-NO of CV-1 is deviated  F-CV-OUT-F-NO is the deviation variable of F-CV-FC in CV-1  F-CV-OUT-F-NO is the deviation variable of F-CV-SC in CV-1  F-CV-INT-F-NO is the deviation variable of F-CV-FC in CV-1  F-CV-INT-F-NO is the deviation variable of F-CV-SC in CV-1  F-CV-IN-F-NO is the deviation variable of F-CV-FC in CV-1  F-CV-IN-F-NO is the deviation variable of F-CV-SC in CV-1</p> <p>deviation-up-umal-list  F-CV-FC is the immediate malfunction of F-CV-CTF in the CV-1  F-CV-FC is the immediate malfunction of F-CV-SNF in the CV-1  F-CV-SC is the immediate malfunction of F-CV-OC in the CV-1  F-CV-SC is the immediate malfunction of F-CV-OP in the CV-1  F-CV-SC is the immediate malfunction of F-CV-LO in the CV-1</p> <p>ad-up-mal-list  FR ==&gt; MATERIAL_RELEASE in PIPE-4  FC ==&gt; PRESSURE_BUILDUP in CV-1  SC ==&gt; PRESSURE_BUILDUP in CV-1</p> <p>ad-up-acc-list  MATERIAL_RELEASE + flammable material ==&gt; FIRE in PIPE-4</p> <p>ad-up-safe-list  FR ==&gt; DT in PIPE-4</p> <p>deviation-down-list  B-PIPE-INT-F-NO is the cause of B-PIPE-OUT-F-NO in PIPE-4  B-SETTLER-IN-F-NO of SETTLER-1 is downdeviated  B-SETTLER-IN-F-NO is the cause of B-SETTLER-INT-L-NO in SETTLER-1  B-SETTLER-INT-L-NO is the cause of B-SETTLER-OUT-F-NO in SETTLER-1</p> <p>ad-down-d-list  NO L =&gt; RUNNING_DRY in SETTLER-1</p> <p>ad-down-safe-list  NO L =&gt; LI or LC in SETTLER-1</p>
---

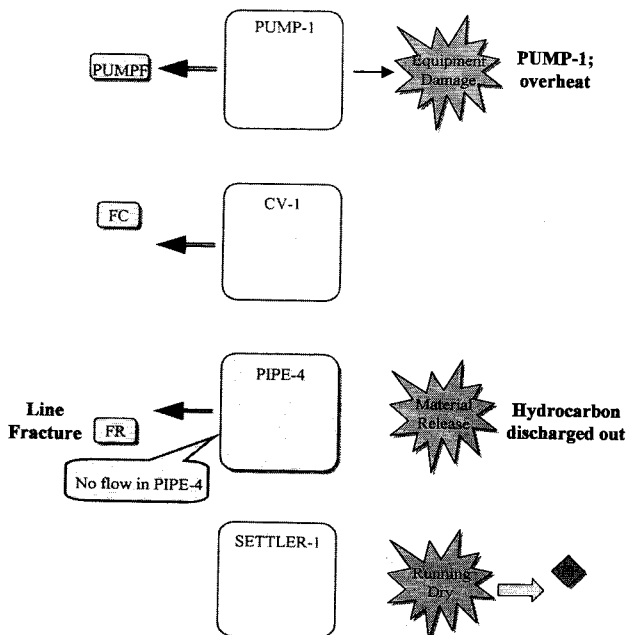


Fig. 10. No flow in pipe-4: Conventional HAZOP's result.

화학공학 제37권 제3호 1999년 6월

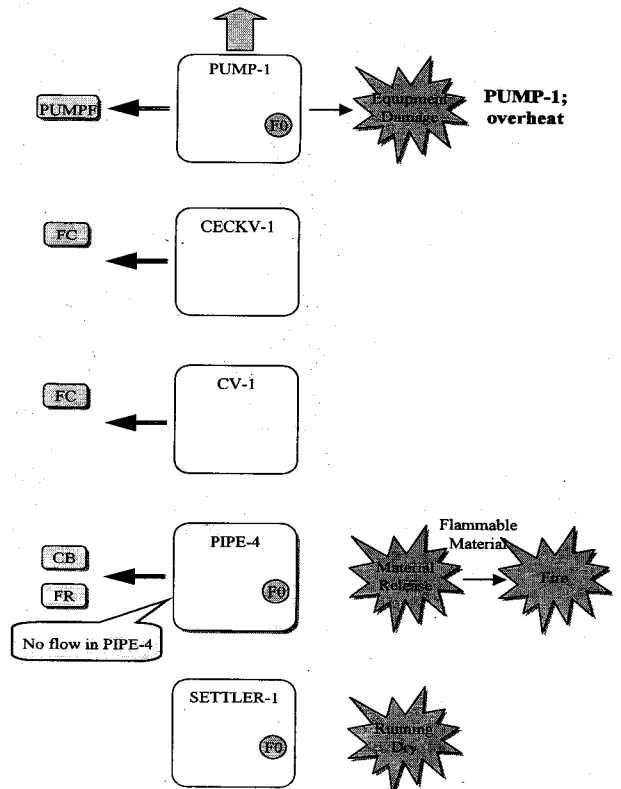


Fig. 11. No flow in pipe-4: HAZOExpert's result.

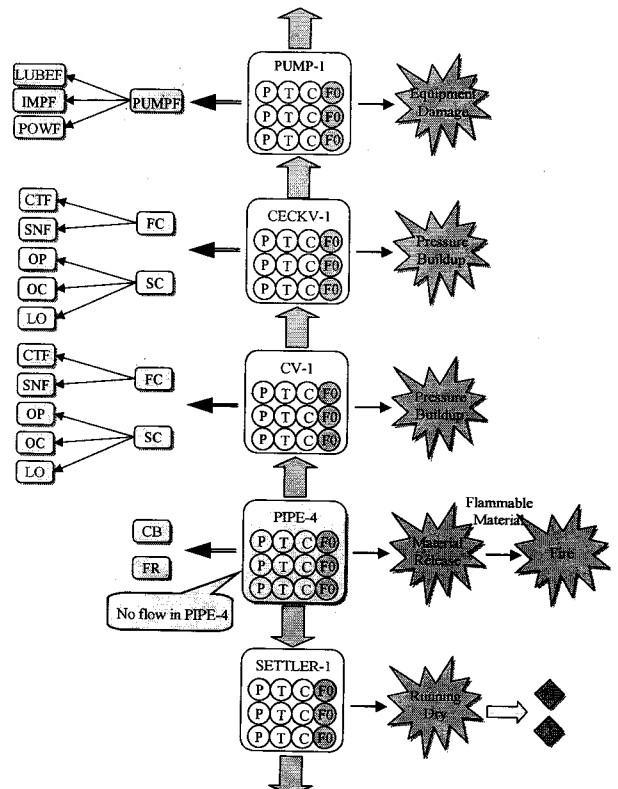


Fig. 12. No flow in pipe-4: AHA's result.

어지기 쉽다. 또한 모든 경우를 확인한다는 것은 실제로 거의 불가능하다. 그러나 자동화 시스템으로는 이런 단점을 충분히 극복할 수 있다.



## 6. 결 론

본 연구는 기존의 안전성 평가의 단점 및 미진한 부분을 보완하고 좀 더 향상된 안전성 평가의 자동화를 위해 다중 모델 접근 방법을 사용하였고, 세 가지의 알고리즘을 통해 평가 방법의 다양성과 보완을 꾀하였다. 그리하여 안전성 자동 분석 시스템인 AHA를 rule과 procedure, frame, class 등을 통해 구현하였다.

이 시스템의 특징으로서는 장치지식베이스에서는 공정장치를 변수와 기능으로 분리하여 근본기능이상과 직접기능이상에 이르는 malfunction hierarchy를 구성하였고 변수는 위치에 따라 inlet, internal, outlet으로 명확히 구분하여 causal relation을 표현하였고, 물질지식베이스에서는 물리적 성질뿐 아니라 화학적 성질까지 고려하였으며, 구조지식베이스에서는 화학공정을 구성하는 여러 stream들을 그들의 역할에 따라 분류하였고, 이를 통하여 공정 구조를 위험성 분석에 적절한 구조로 다시 구성하였다.

알고리즘은 세 가지로 구분하여 각각의 특성을 명확히 하여 변수이탈에서부터 시작하는 deviation analysis algorithm, 기능이상에서부터 시작하는 malfunction analysis algorithm, 공정 장치의 물리적, 화학적 상태로부터 가능한 모든 사고 형태를 표현하는 accident analysis algorithm으로 나누어 화학공장에서 일어날 수 있는 모든 가능한 정보를 나타낼 수 있도록 하였다. 또한 단순한 사고의 원인, 결과뿐만 아니라 사고에 이르기까지의 event combination을 제시함으로써 assessment에 바로 활용할 수 있는 자료를 제시해 준다.

## 감 사

본 연구는 포항공과대학교 지능자동화연구센터를 통한 한국과학재단 우수연구센터 지원금에 의한 것입니다.

## 참고문헌

1. Chae, H. Y.: *Korean J. Chem. Eng.*, **11**, 153(1994).
2. Lee, B. W.: Master Dissertation, Seoul National Univ., Seoul, Korea (1995).
3. An, D. M. and Hwang, K. S.: *HWAHAK KONGHAK*, **34**, 727(1996).
4. Suh, J. C.: *Loss Prevention in the Process Industries*, part 1 & 2, **10** (2), 113(1997).
5. NFPA: NFPA Code 49 Hazardous Chemical Data, National Fire Protection Association(1991a).
6. NFPA: NFPA Code 325M Fire Hazard Properties of Flammable Liquids, Gases, and Volatile Solids, National Fire Protection Association(1991b).
7. Ellis Horowitz: "Fundamentals of Data Structures in C", Computer Science Press(1993).
8. Lees, F. P.: "Loss Prevention in the Process Industries", Butterworth, London(1980).
9. AIChE: "Guidelines for Hazard Evaluation Procedures", 2nd ed., AIChE, New York(1992).
10. Black, J. M. and Ponton, J. W.: Proceedings of IFAC Workshop, 227(1992).
11. Catino, C. A. & Ungar, L. H.: *AIChE Journal*, **41**(1), 97(1995).
12. Shum, S. K., Davis J. F., Punch W. F. and Chandrasekaran, B.: *Comput. Chem. Eng.*, **12**, 27(1988).
13. Venkatasubramanian, V. and Vaidhyanathan, R.: *AIChE J.*, **40**(3), 496 (1994a).
14. Venkatasubramanian, V. and Vaidhyanathan, R.: PSE-94 Proceedings (1994b).