

Fault-consequence digraph를 이용한 전문가 시스템의 지식 베이스 구현

윤 병 석^o, 오 전 근, 윤 인 섭

서울대학교 화학공학과

본 연구에서는 화학공정의 자동화와 관련된 실시간 나프타 분해로 이상진단 시스템의 한 부분을 구성하고 있는 지식 기반의 구현을 위하여 공정변수 상호간의 관계를 나타낸 signed directed graph (SDG)를 기반으로 이상에서 과급되는 증상들을 순서대로 나타내는 fault-consequence digraph(FCD)를 지식모델로 사용했고, 이를 사고사례를 이용하여 검토해본 결과 초기에 정확한 이상원인 후보를 찾아 내므로써 이상진단 전문가 시스템의 지식모델로 적합함을 알 수 있었다.

1. 서 론

화학공장의 규모가 커지고 공정이 복잡해짐에 따라 이상 발생에 따른 신속하고 정확한 대응이 필요해지고 있다. 지금까지 이상진단에 사용되어온 정보에는 alarm의 종류, 관련된 공정변수의 값, 조업자의 그 공정에 대한 지식 등이 있다. 즉 이상이 발생하면 관련된 공정변수의 alarm이 울리게 되고 이 alarm과 다른 공정변수의 변화된 값을 이용하여 조업자는 경험법칙과 불리적인 인과관계를 기반으로 그 원인을 찾게된다. 그러나 조업자에 의한 이상진단은 작업조의 변경, 휴가, 신규사업의 전개에 따른 경험 많은 조업자의 이동, 조업자의 실수 등에 따른 오류가 예상된다. 그러므로 computer를 이용한 공정의 감시와 이상진단 방법이 요구되고 있다. 이상진단 전문가 시스템에 필요한 지식기반을 구현하기 위한 지식모델로서는 경험법칙을 계층구조로 나타낸 모델[3], 공정변수 상호간의 관계를 나타낸 signed directed graph(SDG)를 이용한 모델[2], artificial neural network를 이용한 모델[1]등 이 있다. 이중 SDG를 이용한 방법은 공정변수 상호간의 관계를 +, - 의 관계로 나타낸 것으로 이것을 근거로 원인에서 시작되는 simulation tree를 작성하여 실제의 data와 비교하여 이상원인을 찾는 방법이다.

본 연구에서는 실제의 나프타 분해로 공정에 대한 SDG를 작성하고 이를 근거로 fault-consequence digraph (FCD)를 작성하여 real time 이상진단 전문가 시스템의 지식기반구현에 필요한 지식모델로 사용하였다. 이때 FCD를 이용한 이상진단에는 증상발생의 순서와 수학적인 모사, 확률 등은 사용하지 않았고 하나의 이상이 발생해서 그 원인을 찾는 동안에는 다른 이상이 발생하지 않는다는 것을 가정했고 sensor 고장인 경우에도 그 sensor에 나타나는 값을 그대로 이용하였으며 하나의 같은 이상원인에서

반대 방향으로 나타나는 애매한 증상들은 진단과정에서 고려하지 않았다.

2. 이 론

2.1 Signed directed graph

SDG의 node는 상태변수, alarm 조건, 이상 원인을 나타내고 화살표는 node간의 원인-결과 관계를 나타내며 이때 (+), (-) 는 node간의 관계의 방향을 나타낸다. SDG의 작성과 이를 이용한 rule의 생성은 다음과 같다.

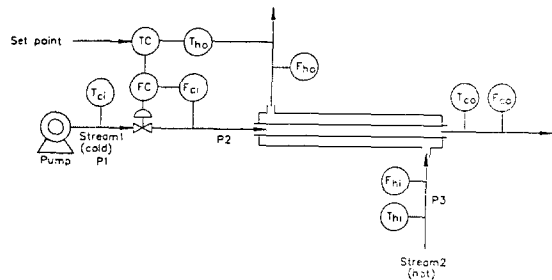


그림 1 Heat exchanger with cascade control loop

그림 1 은 stream 2의 출구 온도를 일정하게 유지시켜 주기 위해서 냉매의 유량을 조절해 주고 있는 열교환기이다. 냉매 유량 제어기는 stream 2의 출구온도를 측정해서 set point와 비교한 다음 제어기 valve에 적절한 output을 내려주고 그 결과로 냉매의 유량을 조절해서 온도를 원하는 값으로 유지한다. 이때 output의 증가는 냉매의 유량의 감소를 나타낸다.(fail-to-open system) pump의 회전수는 공정의 다른 변수에 의해서 영향을 받지 않고, 출구 온도의 set point는 조업자가 지시하게 되어 있으며 Fci와 Tno를 제외한 나머지 sensor들의 이상은 공정에 영

향을 미치지 않는다. 이 system에서 얻을 수 있는 정보는 T_{ci} , F_{ci} , T_{ho} , F_{ho} , T_{co} , F_{co} , F_{hi} , T_{hi} , S_F , O_F , V_F 등이다. sensor는 제어기에 직접적으로 영향을 미치는 제어 sensor와 단지 어떤 값을 측정하고 나타내 주는 지시기 역할을 하는 sensor로 구분할 수 있다.

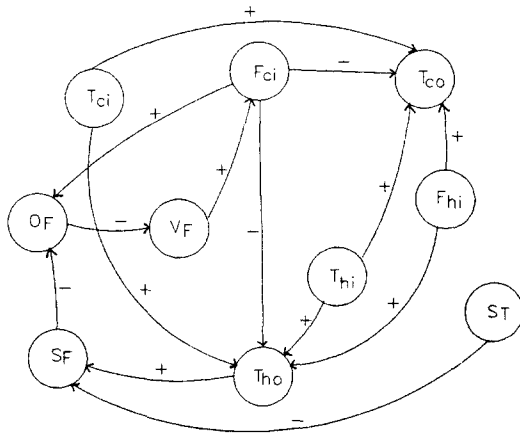


그림 2 Digraph of heat exchanger system

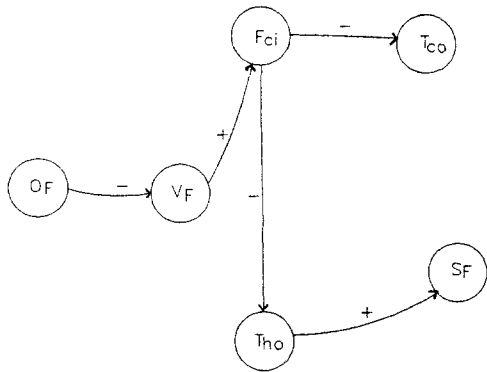


그림 3 Reduced digraph, $O_F=+1$ as root node

그림 2에서 예로 선택된 node는 $O_F=+1$ 로서 이것은 제어기 output의 양의 방향으로의 이상을 나타내는 것이다. SDG를 이용하여 생성된 simulation tree는 그림 3과 같다. $O_F=+1$ 에서 시작해서 도달할 수 없는 node는 T_{ci} , S_T , F_{hi} , T_{hi} , 등이며 작성된 rule은 다음과 같다.

- $(T_{ci}=0)$ and $(S_T=0)$ and $(F_{hi}=0)$ and $(T_{hi}=0)$
- and $(mO_F V_F)$
- and $(pV_F F_{ci})$
- and $(mF_{ci} T_{co})$
- and $(mF_{ci} T_{ho})$
- and $(pT_{ho} S_F)$

여기에서 p, m, z의 진리표는 다음과 같다.

표 1 Truth Table

A \ B		A \xrightarrow{p} B		
		1	0	-1
1	F	T	F	
0	F	T	F	
-1	F	T	F	

A \ B		A \xrightarrow{m} B		
		1	0	-1
1	T	T	F	
0	F	T	F	
-1	F	T	T	

A \ B		A \xrightarrow{z} B		
		1	0	-1
1	F	T	T	
0	F	T	F	
-1	T	T	F	

위와 같이 작성된 rule들은 이상진단에 이용될 수 있으나 원하는 증상이 중간에서 감지되지 않는 경우에 그 이상원인에 대해서 정확한 진단을 내릴 수 없다.

2.2 Fault-consequence digraph

(1) FCD의 필요성

어떤 공정의 이상을 찾아내기 위해서는 그 공정에서 얻을 수 있는 모든 정보를 활용해서 미리 정의해둔 이상에 대응하는 증상들을 발생하는 순서를 따라서 표시하는 방법이 유용할 수 있다. FCD는 예상되는 사고로부터 어떤 변수에 어떤 형태로 영향을 미치는가를 표시한 점에서는 SDG와 유사하다고 할 수 있으나 SDG의 경우 앞의 상황이 어떤 사고에서 부터 파급되어온 것인지에 따라서 그 뒤의 상황이 달라지게 되는 경우를 구별하기 어렵고 detection상에서 FCD에서 순서대로 정의한 증상들 중에 하나만 감지하지 못하더라도 정확한 진단을 내릴 수 없다. 그러므로 각각의 사고에서 부터 그 다음 상황을 정의한 FCD를 작성하고 새로운 표현방법을 사용하여 실제의 공정에서 얻을 수 있는 data와 비교해서 어떤 사고가 발생했는지를 찾아내는데 이용할 필요가 있다.

(2) FCD의 작성

i) 고정에서 얻을 수 있는 변수를 수집하고 그 변수를 중심으로 각각의 변수끼리의 인과관계를 나타내는 SDG를 작성한다.

ii) 그 공정에서 발생할 수 있는 모든 이상을 정의한다. 이때 이상은 그 공정에 직접적으로 영향을 주는 것으로 정의한다. 즉 control valve의 고장을 정의할때 그 valve가 닫히는 상태의 고장인가 열리는 상태의 고장인가

만을 정의하고 고장의 원인등은 더이상 구별하지 않는다.

iii) 이상들 중에서 공정변수에 같은 영향을 줄수 있는 것들을 이상군으로 만든다.

iv) 작성된 SDG를 참고로 하여 각각의 이상군에서 시작하는 FCD를 작성한다. 이때 그 발생순서에 따라서 화살표 방향으로 증상들을 나타낸다. 이때 변수들이 (+)값을 나타낼지 또는 (-)값을 나타낼지를 표시한다. 이때의 (+), (-)는 그 변수의 값이 그때까지 조업되어온 값에서의 변동 상태를 나타낸다. 또한 어떤 특정한 조건이 만족될 때만 그 다음의 상황이 전개되는 경우에 그 조건의 만족 여부를 가리기 위해서 조건부(condition gate)를 해당되는 원인 결과 증상 사이에 나타낸다. FCD 작성시 사용되는 기호는 다음과 같다.

○; 공정상에서 그 변수의 경향이 바뀌는 경우를 나타낸다.

□; computer로는 들어오지 않지만 실제 공정변수값이 변할

때 사용된다.(예:sensor 고장)

◇; 임의의 조건이 만족하는 경우에만 그 다음 상태로 전개됨

◇; 그 뒤로 다른 상황이 전개된다.

□; 이상군을 나타낸다.

→; 다음 상황이 FCD에 나타난 부호의 방향으로 전개된다.

....→; 다음 상황이 FCD에 나타난 부호와 반대방향으로 나타난다.

—; 양쪽에 연결된 변수들의 순서가 거의 비슷하거나 구별하기

어려울 때 사용된다.

(3) FCD의 작성 예

그림 1 의 예제에서 생각할 수 있는 이상의 종류는 다음과 같다.

1. pump의 고장(+, -)
2. 제어기 sensor 고장(+, -)
3. 독립 sensor 고장
4. valve 작동 불능(+, -)
5. 제어기의 output의 오지시(+, -)
6. P₁, P₂, P₃, P₄, P₅ 의 leaking
7. stream 1 pipe의 plugging
8. stream 2 pipe의 plugging
9. fouling

10. coolant temp. 변화

11. hot stream temp. 변화

12. hot stream 유량 변화

13. 출구 온도 set point 오지시

이 예제와 관련된 SDG는 그림 2 와 같고 FCD의 작성은 다음과 같다. 여기에서는 많은 이상중에서 냉매 공급 pump의 성능이 저하되는 경우와 제어기 sensor 고장에 대한 예를 들었다.

i)

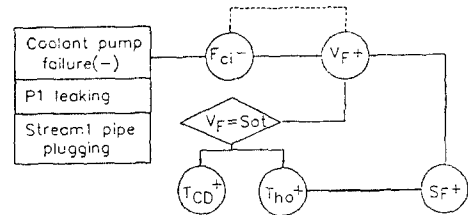


그림 4 Coolant pump failure

냉매의 pumping 되는 양이 줄어들면 제어기에 의해 valve가 열리게 되어서 유량을 일정하게 유지해주지만 그 정도가 심하면 제어기가 조절할 수 없어서 유량이 줄어들게 되며 이에 따라서 T_{co}와 T_{ho}가 증가한다. 이때 T_{co}의 증가에 따라서 S_p가 증가하게 되고 이에따라 냉매 유량 제어기의 output은 감소하게 되지만 더이상 valve가 열릴 수 없으므로 증상이 없어지지 않는다. 냉매의 pumping양이 너무 많을때는 위의 그림과 반대 부호의 그림을 얻을 수 있다. P1 leaking이나 stream 1의 pipe plugging 인 경우에는 위와 같은 현상이 나타나게 되는데 이 이상들을 정확히 구분할 수는 없다. 또한 pump의 성능 변화에는 많은 원인이 있으나 여기서는 더이상의 원인 진단은 생각하지 않는다. 그림 4 를 원인-결과식으로 나타내면 다음과 같다.

((of n)

fci-of- of-vf+ (vfat.con vf+tcd+ vf+tho+ tho+st+ st+sf+))

위에서 냉매의 유입량이 줄게되면 제어기의 output은 플럼없이 감소할 것이라고 생각되므로 다른 증상들과 구분해서 표현하였다. fci-of-는 fci가 감소하고 이에따라 of가 감소한다는 것을 나타내고있고, vfat.cond은 유량 제어 valve가 더 이상 움직일 수 없는 상황이 발생했을 때 그 다음의 상황이 전개되는 것을 나타내고 있다. 이 조건이 만족되지 않으면 팔호안의 사실은 성립되지 않는다. 그림 4 에서 보면 valve가 열림에 따라 유량이 증가하는 관계가 나타나 있으나 제어기가 작동하더라도 이미 감소된

유량을 이전의 조건보다 더 증가시킬 수는 없으므로 원인-결과식에는 나타나지 않는다. 또한 set point의 증가에 따른 output의 감소는 output의 감소가 이미 나타나 있으므로 중복을 피하기 위해 나타나지 않는다.

ii)

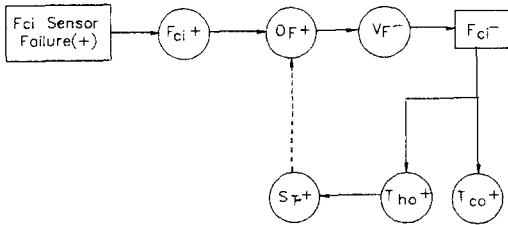


그림 5 Coolant inlet flowrate sensor failure

이 예제는 제어기와 관련된 sensor의 고장이 공정에 어떻게 영향을 주는가를 보여주고 있다. F_{ci} sensor가 실제값보다 높은 값을 가리키게 되었을 때 제어기는 valve를 닫아주게 되고 이때 실제의 F_{ci} 는 감소하게 되며 이에 따라 T_{co} , T_{ho} , S_F 가 증가하게 되며 S_F 증가에 따라 O_F 는 다시 감소하는 경향을 갖게 된다. 이때 F_{ci} 는 증가치를 나타내지만 그 뒤의 상황은 F_{ci} 가 감소했을 때의 상황을 나타낸다. 이때 유량의 감소에도 불구하고 그 다음의 상황은 반대로 나타나므로 이 경우에는 다른 원인 후보들과 쉽게 구별할 수 있다. 그림 5를 원인-결과식으로 나타내면 다음과 같다.

((fci p)

fci+of+ of+vf- vf-tco+ vf-tho+ tho+st+ st+sf+)

위의 예에서 살펴 보았을 때 각각의 이상에 따라서 나타나는 증상의 종류와 순서와 방향(+, -)이 달라지게 되므로 증상에서 얻을 수 있는 정보들로부터 이상상태에서 어떤 이상이 발생했는지 알아볼 수 있다. 이때 어떤 이상이 발생했을 때 다른 조건들은 정상상태에 있다고 가정한다. 여러가지 이상이 짧은 시간에 함께 발생할 때는 이상을 진단할 수 없다.

2.3 나프타 분해로 이상진단 전문가 시스템(OASYS)의 지식기반 구현

(1) 나프타 분해로 공정

나프타 분해로 공정은 하나의 커다란 열교환기라고 생각할 수 있다. 나프타 분해로는 크게 나프타와 분해로 내에서 과도한 coking을 방지하기 위한 희석 증기(dilution steam)로 이루어진 feed 부분, 분해로의 열원인 버너로 연료를 공급해 주는 연료 가스 부분, 분해된 나프타를 식혀주고 증기를 발생시키는 steam drum 부분 등의 세부 분으로 구성되어 있다.

feed pump로 부터 공급된 나프타는 희석 증기와 섞여서 대류 부분(convection zone)에서 예열된 다음 짧은 시간동안 분해로의 복사 부분(radiation zone)을 통과하면서 분해되고 TLX(transfer line exchanger)에서 식혀져서 다음 공정으로 나가게 된다. 이때 TLX에는 steam drum에 저장된 BFW(boiling feed water)가 냉매로 작용한다. 특히 이 steam drum부분에 이상이 생겨서 나프타가 원하는 온도로 식혀지지 않으면 다음 공정에 치명적인 피해를 주게 된다. 나프타 분해로 공정에는 DMC(dynamic matrix control) logic이 이용되는데 이는 나프타의 유량, 희석증기의 유량, 연료 압력등의 set point를 공정상태에 따라서 자동적으로 변경한다. DMC의 작용으로 나프타 분해로 공정의 이상진단에 어려움이 있다.

(2) 진리 유지체계 (truth maintenance system)

지식기반의 사실들 간에는 dependancy가 성립될 수 있다. 즉 어떤 사실은 그 사실을 지지하는 다른 어떤 사실들이 참일 때 항상 참이 될 수 있다.

TMS는 사실을 두 가지로 구분한다. 근본 사실(primary fact)은 그 사실의 참여부가 다른 사실의 참여부에 전혀 영향을 받지 않는 사실이고, 연역 사실(deduced fact)은 그 사실의 참여부가 다른 사실의 참여부에 의존되는 사실이다. TMS는 또한 어떤 사실이 선언 되었을 때 그 이유를 설명하고 추론세계에 모순이 발생 되었을 때 그 모순의 원인을 보여주기도 한다.

OASYS에서는 TMS를 사용하여 이상원인에 대한 사실을 선언하고 그 사실들을 실제 공정 data와 비교해서 그 이상원인의 적합성을 판단하게 된다. 또한 TMS의 설명기능을 이용하여 사실선언과정과 모순원인을 보여줄 수 있으므로 조업자의 체계적인 지식정립에도 도움을 줄 수 있다.

(3) FCD는 OASYS-TMS 지식 기반 (knowledge base) 안에 있는 TMS Unit의 fcd-slot 내에 저장되게 된다. 저장되는 형태는 어떤 특정한 이상에 대하여 나타나는 공정변수들의 관계를 원인-결과식으로 표현한다. 하나의 원인-결과식은 각각 하나의 unit를 구성하고 있으며 특정한 slot안에 원인변수와 결과변수를 나타낼 수 있는 형태로 지식기반에 저장한다. 지식기반에 저장된 FCD는 symptom tree analysis에서 제공된 가능한 원인후보들에 대해서 TMS를 이용하여 그 원인후보들의 발생을 가정하여 일어난 증상들을 모사하고 실제공정에서 받은 data와 비교해서 서로 상반되는 현상들이 일어나는지의 여부를 따지게 된다. 이때 fcd-slot에는 어떤 이상이 발생하게 되면 반드시 일어날 것이라고 생각되는 증상(primary effect)을 다른 원인 결과식(secondary effect)들과 구분하여 표현하였으므로 반드시 일어나야할 증상이 실제 data와 모사된 사실 간에 모순이 되면 이 원인후보는 제외가 된다. 이것은 물리적인 인과관계를 따져서 FCD를 작성할 때 조업자의 경험법칙에 따라서 어떤 증상을 primary effect로

표현하면 진단시 바른 판단을 내리는데 도움을 줄 수 있다는 것을 나타낸다. 그림 6 은 희석 증기(dilution steam)의 유량 sensor가 실제값보다 낮은 값을 나타내는 이상에 대한 FCD를 보여주고 있다.

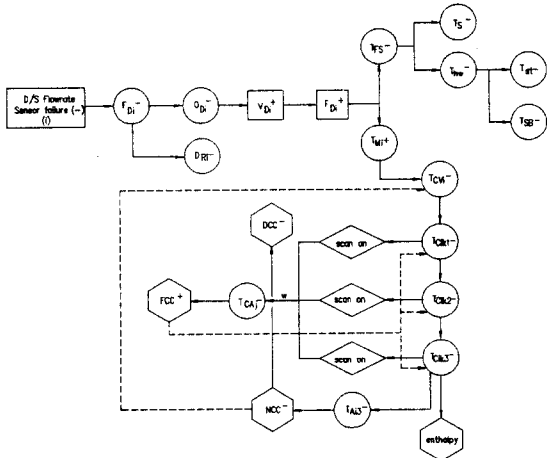


그림 6 FCD for dilution steam flowrate sensor failure

그림 6의 FCD는 다음과 같이 표현될 수 있다.

((dflowrate n)
 (fdi-dri- fdi-odi- odi-tfs- odi-tmi+ tmi+tcvi- stackt-
 tcvi-telik- coil- fcc+ tca3i-sni- sni-sdi- ncci-))

stackt-, coil-, fcc+, ncci- 등은 여러 원인-결과식을 포함하고 있다.

TMS의 사용에 따라서 이 시스템의 사용자는 어떤 원인 후보에 대하여 예상되는 증상을 알아볼 수 있으며 또한 어떤 사실이 어떤 원인으로 부터 성립되는지를 단계적으로 살펴볼 수 있으며 실제 data와 모순이 생겼을 때 그 모순의 원인을 찾아 볼 수도 있다. 그림 7 은 어떤 원인 후보에 대한 모사 사실을 보여주고 있고 그림 8 은 모순이 발생한 원인 후보에 대해서 그 원인을 설명해주는 그림이다.

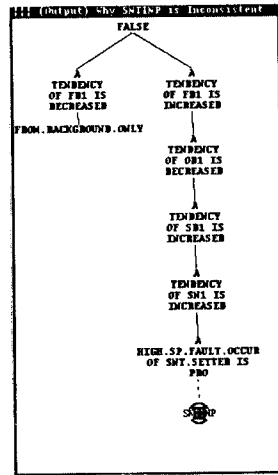


그림 8 Explain of conflicted world

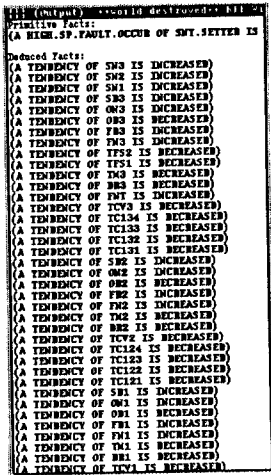


그림 7 Simulated facts

3. 결론

본 연구에서는 나프타 분해로를 대상공정으로 314개 이상을 정의했고 이에 대한 FCD를 작성한 후 SUN3/260에 KEE를 사용하여 지식기반을 구축하고 226개의 증상을 이용하여 증상 트리 모델(symptom tree model)에서 제시되는 원인 후보에 대한 모사를 했다. 이때 실제 사고 사례와 가상사례의 data는 PC/AT를 통해서 SUN3/260에 제공된다. 그 결과 고장의 발생초기에 실제 예상되는 이상 원인을 증상 트리 모델에서 제시된 것보다 범위를 좁혀서 찾아내는데 성공하였다. 그러나 앞으로 FCD 지식 모델에 있어서 제어에 따른 반대 증상의 발생 문제의 표현과 증상의 순서를 진단에 이용할 수 있는 방법의 연구가 필요하다.

이와같이 개발된 이상진단 전문가 시스템은 실시간 공정에서 이상의 원인을 신속하고 정확하게 찾아낼 수 있으며, 또한 작업자의 훈련과 공정 관리에도 이용될 수 있고, 이에 따라 작업성 향상에 기여할 수 있다.

감사

본 연구의 공정지식 획득에 많은 도움을 주신 (주)유공의 여러분들께 깊은 감사를 드립니다.

참고문헌

1. J. C. Hoskins and D. M. Himmelblau, "Artificial Neural Network Models of Knowledge Representation in Chemical Engineering", *Computer Chem. Engng.*, Vol. 12, No. 9/10, 1988
2. M. A. Kramer and B. L. Palowich, "A Rule-Based Approach to Fault Diagnosis Using the Singed Directed Graph", *AICHE J.*, Vol. 33, No. 7, 1987
3. S. K. Shum, J. F. Davis, W. F. Punch and B. Chandrasekaran, "An Expert System Approach to Malfunction Diagnosis in Chemical Plants", *Computer Chem. Engng.*, Vol. 12, No. 1, 1988
4. S. H. Rich, and Venkat Venkatasubramanian, "Causality-Based Failure-Driven Leading in Diagnostic Expert Systems", *AICHE J.*, Vol. 35, No. 6, 1989
5. S. H. Rich, and Venkat Venkatasubramanian, "Model-Based Reasoning in Diagnostic Expert Systems for Chemical Process Plants", *Comput. Chem. Eng.*, Vol. 11, No. 2, 1987
6. T. S. Ramesh, S. K. Shum and J. F. Davis, "A Structed Framework for Efficient Problem Solving in Diagnostic Expert Systems", *Computer Chem. Engng.*, Vol. 12, No. 9/10, 1988
7. Venkat Venkatasubramanian and S. H. Rich, "An Object-Oriented Two-Tier Architecture for Integrating Compiled and Deep-Level Knowledge for Process Diagnosis", *Comput. Chem. Engng.*, Vol. 12, No. 9/10, 1988