

# 추론엔진을 위한 ECBM의 설계 구현

신정훈<sup>†</sup> · 오명륜<sup>††</sup> · 오광진<sup>††</sup> · 이양원<sup>†††</sup> · 류근호<sup>††††</sup> · 김영훈<sup>††††</sup>

## 요 약

1970년대 후반에 제안된 전문가 시스템은 인공지능의 한 분야로서, 인간의 사고방식을 모방함으로써 다양한 분야에서 야기되는 문제들을 해결해준다. 대부분의 전문가 시스템은 추론엔진과 지식베이스등과 같은 많은 요소들로 구성된다. 특히 전문가 시스템의 성능은 추론엔진의 효율성에 의해 좌우된다. 이러한 추론 엔진은 지식베이스가 구축될 때, 가능한 한 적은 제약성을 가져야 함은 물론, 다양한 추론 방법을 제공해야 한다는 특징을 갖고 있어야 한다.

이 논문에서는 지식 영역과 추론 방식에 대한 범용성을 제공하는 추론 엔진을 설계 및 구현하였다. 이를 위해 추론 방식은 사용자에게 의해 전향추론과 후향추론 및 직접추론이 선택적으로 수행된다. 또한 목표 영역에서의 지식 획득을 위한 쉬운 표준화와 모듈화를 가능케하는 생성 규칙을 사용하였을 뿐만 아니라 확장된 CBM을 통해 지식 베이스를 구축하였다. 아울러, Rete 패턴 매칭과 ECBM을 이용한 추론 엔진간의 성능분석을 수행하였다.

## Design and Implementation of the ECBM for Inference Engine

Jeonghoon Shin<sup>†</sup> · Myeonryoon Oh<sup>††</sup> · Kwangjin Oh<sup>††</sup> · Yang Weon Rhee<sup>†††</sup> ·  
Keun Ho Ryu<sup>††††</sup> · Young Hoon Kim<sup>††††</sup>

## ABSTRACT

Expert system is one of AI area which was came out at the end of 1970s. It simulates the human's way of thinking to give solutions of problem in many applications. Most expert system consists of many components such as inference engine, knowledge base, and so on. Especially the performance of expert system depends on the control of efficiency of inference engine. Inference engine has to get features; first, if possible to minimize restrictions when the knowledge base is constructed second, it has to serve various kinds of inferencing methods.

In this paper, we design and implement the inference engine which is able to support the general functions to knowledge domain and inferencing method. For the purpose, forward chaining, backward chaining, and direct chaining was employed as an inferencing method in order to be able to be used by user request selectively. Also we not only selected production system which makes one ease staradization and modulation to obtain knowledges in target domain, but also constructed knowledge base by means of Extended Clause Bit Metrics(ECBM). Finally, the performance evaluation of inference engine between Rete pattern matching and ECBM has been done.

\*이 연구는 1996년도 삼성 중공업 중앙 연구소의 연구비 지원으로 이루어졌음.

† 정 회 원: 이리농공전문대학 전자계산과

†† 준 회 원: 충북대학교 대학원 전자계산학과

††† 중 심 회 원: 군산대학교 컴퓨터과학과

†††† 중 심 회 원: 충북대학교 컴퓨터과학과

††††† 정 회 원: 삼성중공업 중앙연구소

논문접수: 1997년 9월 3일, 심사완료: 1997년 10월 30일

## 1. 서론

전문가 시스템은 1970년대 후반에 인공지능의 한 분야로써 알고리즘이나 일반적인 탐색방법보다는 주어진 지식을 이용하여 많은 응용 문제들을 해결하여 해를 제시하는데 있어 인간이 사고하는 방식을 모방하는 방법을 이용하는 시스템으로 대두되기 시작하였다. 전문가 시스템은 일반적으로 전문가의 지식을 필요로 하는 상당수의 문제 영역(problem area)에서 효과적인 것으로 증명되어 왔다[26]. 대부분의 전문가 시스템은 인간의 사고 방식을 모방하여 기계화한 추론엔진, 전문가의 지식을 부호화(symbolic)하여 저장해 놓은 지식베이스를 주축으로 기타 필요한 기능들에 따라 추가하여 구성된다.

전문가가 갖고 있는 지식을 이용하여 어떠한 상황에 대한 판단을 하는 과정을 추론이라 하며, 일련의 추론 과정을 컴퓨터가 수행할 수 있도록 설계·구축되어진 것을 추론엔진이라 한다. 이러한 추론엔진의 구성은 추론시 이용되는 지식표현과 그 구축형태에 따라 변화될 수 있으며, 운영방법의 제한이 있을 수도 있다. 이는 추론엔진과 지식베이스의 지식 구축 형태가 서로 밀접한 관련이 있음을 이야기 한다. 추론엔진의 성능은 일반적으로 매처(matcher)에 의해 좌우된다. 최근까지의 성능 향상의 노력은 매처에 이용될 알고리즘과 방법론을 중심으로 연구되어 왔다.

최근에 실세계의 도메인은 매우 복잡하고 다양하며, 그 성격이 뚜렷하지 않아 애매한 경우가 많아졌다. 이러한 다양한 도메인마다 지식을 획득하여 지식베이스를 구축하고 획득된 지식을 바탕으로 추론을 하여 원하는 해(solution)를 출력하는 전문가 시스템을 각기 구성하는 것보다 범용성을 갖는 전문가 시스템을 구축하여 사용자의 요구사항에 맞게 사용할 수 있는 것이 더 요구되고 있다. 이러한 요구 사항을 만족하려면 지식의 습득에 용이성과 습득된 지식에 구축의 용이성, 그리고 다양한 성격의 도메인에 대한 융통성이 가능한 지식 구축 기법이 필요하고, 이에 특성에 맞는 매처를 가진 추론엔진이 필요하다.

이 논문은 범용성을 갖는 전문가 시스템 셀을 구축하는 주요 구성 요소 중에서 지식베이스 구축형태인 ECBM(Extended Clause Bit Matrix)를 제시하여 이에 맞는 매처를 갖는 추론엔진을 구축하였다. 즉 추론의

기본이 되어지는 지식을 지식베이스로 구축하는 방법에서의 개선과 이에 적합한 매처를 설계하는 것을 통해 추론엔진과 지식베이스의 양측면에서의 성능향상을 도모하는 것이 목적이다. 그러기 위해서, 표준화와 모듈화 뿐만 아니라[16], 유연성을 제공해 주어 널리 이용되는 방법인 생성규칙[3]을 지식표현 방법으로 선택하였다. 최근 널리 쓰이고 있는 Rete, TREAT와 같은 트리 구조는, 추론 시에 이용할 지식표현 방법과 추론운영 방식에 제한을 둘 수 있으며, 일반적인 도메인에 쉽게 적용할 수 없는 단점으로 인해 연구의 목적에 적절하지 않아, 여기에서는 선택하지 않았다. 따라서, 이 논문에서는 특정 도메인과 특정 추론운영 방식에 좌우되지 않는 범용적인 특성을 갖는 지식표현 구조인 비트 매트릭스를 선택하고, 이의 단점을 개선하고 주어진 지식 영역에 적합한 지식표현 구조인 ECBM을 제안하며, ECBM을 기반으로 전향추론, 후향추론을 수행하는 시스템을 설계·구현 한다.

이 논문은, 1장에서 연구의 배경과 목적을 간략하게 설명한 후, 2장에서 필요한 관련연구를 살펴보고, 3장에서는 추론을 위한 지식베이스의 구현방법으로 ECBM을 제안하고 ECBM을 이용해 구축한 추론엔진을 설명하며, 구축된 추론엔진의 성능에 대한 평가는 4장에서 다룬 후, 5장에서는 그동안의 연구에 대한 결론을 내리고 향후 연구과제를 제시한다.

## 2. 관련 연구

지식표현의 종류로는 의미망, 프레임, 그리고 생성규칙 등이 있다. 인간의 사고에 대한 심리적인 모델을 사용하여 지식을 표현하는 의미망(semantic net)은 속성 계승(inheritance)의 장점을 갖는다[11, 19]. 반면에, 모델화를 하기 위해 노드들이 중복되어 그 크기가 방대하고, 지식 수정과 타당성 검증이 어렵고 탐색시간이 많이 걸린다는 단점을 갖고 있다. 미리 정의된 내부 관계들을 갖는 단일 환경에서의 선언적, 절차적 지식을 조합한 자료구조인 프레임(frame)은 의미망과 구조가 유사한 그래프로서 프레임의 이름, 프레임의 부모, 프레임의 슬롯 등으로 구성된다. 프레임의 장점은 속성 계승이지만, 저장공간의 낭비와 탐색속도가 의미망보다 더 느리며, 매우 어렵고 복잡한 추론엔진을 요구하는 단점이 있다[14].

끝으로, 이 논문에서 지식표현 방법으로 선택한 생성규칙(production rule)은 "If A Then B" 형태를 가진 표현 방법으로[1, 10], 누구나 쉽게 접근할 수 있으며, 규칙들이 서로 상호 독립적이며, 수정이 용이하고 쉽게 설명이 가능하고 쉽게 습득할 수 있는 등의 장점을 갖고 있다[4, 14]. 생성규칙은 그 표현을 어떻게 구현하는가에 따라서, 규칙의 결론과 전제의 관계를 노드와 링크의 관계로 직접적으로 그래프 표현을 하는 트리 구조, 규칙의 절들간의 연관관계를 "0"과 "1"로 표현하는 비트 매트릭스 구조, 규칙의 전제와 결론의 연결 고리를 저장하는 관계 리스트 구조 등으로 분류할 수 있다[19].

## 2.1 생성규칙의 구현방법

### 2.1.1 트리 표현

트리 표현은 규칙들의 전제부(premise)와 결론부(conclusion) 사이의 관계를 명시하는 지식표현으로, 규칙을 표현하는 노드와 규칙의 전제부와 결론부 사이의 관계를 나타내는 링크로 구성되며, 규칙트리(rule tree)와 규칙의 절트리(clause tree)로 나눌 수 있다.

규칙트리 표현은 일반적으로 지식베이스의 첫 번째 규칙부터 마지막 규칙까지 탐색을 하면서, 링크로 연결된 모든 다른 규칙들의 전제들과 현재 검토 중인 규칙의 결론들 사이의 연결을 평가한다[18, 19, 20].

절트리 표현은 규칙트리 표현에서 제공해 주지 못하는 추가적인 정보가 필요하다. 규칙의 결론과 전제의 관계 뿐만 아니라, 전제들간의 상호관계를 표현해 주어야 한다.

일반적으로, 트리표현은 규칙이나 절이 다른 규칙이나 절과 어떻게 연관되어 있는가를 보여주는 것이 가능하고, 쉽게 추적이 가능하며, 트리 표현으로 구축되어진 지식베이스를 갖는 추론 과정은 매우 빠르며, 직접 추론경로의 설명을 할 수 있다.

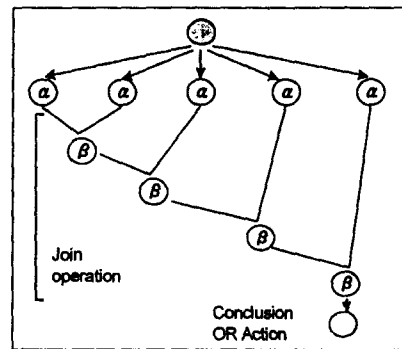
트리 표현 방법에는 Rete과 TREAT 등이 있다.

#### (1) Rete

전향추론 시에 효율적임이 입증된 알고리즘인 Rete는 1970년대 후반에 Forgy[5]에 의해 제안되었으며, 생성규칙을 보편화 시켜준 구현방법으로 다수의 객체에 대한 다수의 패턴매칭 기법이다. Rete 알고리즘은 소개된 이후로 지금까지 생성규칙을 표현하는 좋은

방법으로 인식되어 왔다.

Rete의 구성은 루트노드, 규칙의 전제부와 일치하는 규칙들을 보관하는 알파메모리(alpha memory), 이들 알파메모리를 조건에 맞게 두 개씩 일정 제약을 통해 합하여 보관하는 베타메모리(beta memory), 그리고 생성규칙에 대한 최종 결과 노드인 P노드가 기본요소이며, 그림 1과 같이 구성된다.



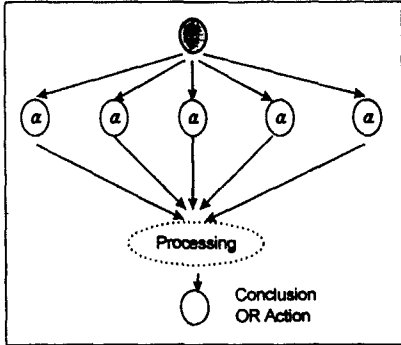
(그림 1) Rete의 예  
(Fig. 1) Example of Rete

이 방법은 많은 규칙들에 대한 패턴 매칭시에는 탐색속도가 매우 빠르지만, 수 많은 알파메모리와 이들을 일정조건에 의해서 결합시켜 생성시킨 베타메모리의 보관에 따른 저장공간의 비효율성이 단점으로 지적되어 왔다.

최근에는 이와 같은 단점을 극복하기 위한 방법으로 Uni-Rete[22]와 같이 특정한 분야에 대해 효율성을 극대화 시킬수 있는 형태로 변형되어 소개되고 있으며, 또 다른 변형된 형태의 알고리즘들에 관한 연구가 계속되고 있다.

#### (2) TREAT

생성규칙을 표현하는 수단으로 Rete가 좋은 패턴 매칭 알고리즘으로 자리잡아왔다. 이에 따라 Rete의 변형된 알고리즘들이 소개되고 있으며, 그 중 하나로서 TREAT를 들 수 있다. 기본구조는 그림 2에서 보여주고 있다.



(그림 2) TREAT의 예  
(Fig. 2) Example of TREAT

TREAT 구성은 Rete의 1단계 분할(sharing)개념은 이용하지만, 2단계 상태저장(state saving) 개념은 없다. 다시 말하자면, Rete 알고리즘 중 두 개의 알파메모리를 조합하여 하나의 베타메모리를 생성·저장하는 단계는 실행하지 않는다. 1단계 수행 후, 추론 시에 필요한 알파메모리들을 조합하는 것으로 그 구성을 마친다.

이러한 구성은 이미 생성되어있는 구조에서의 탐색 속도가 Rete보다 느려[17], Rete의 단점에 대한 대안의 역할보다는 변형된 새로운 패턴매칭 기법의 역할을 수행한다[8].

2.1.2 비트 매트릭스 표현

비트 매트릭스 표현은 생성규칙의 전제부와 결론부를 각각 X, Y축에 위치시킨 2차원 행렬의 구조를 갖는 방법이다.

이러한 지식표현은 첫째, 추적(trace)이 매우 빠르며, 둘째, 구현이 매우 쉽다는 장점을 가진 다중 연결을 표현할 수 있는 자료구조이다. 비트 매트릭스 표현은 규칙 비트 매트릭스(RBM: Rule Bit Matrix), 절 비트 매트릭스(CBM: Clause Bit Matrix)와 저장 효율성을 증대시키기 위해 연구된 축소 CBM(Reduced CBM) 등으로 분류할 수 있다[19].

(1) 규칙 비트 매트릭스

규칙 비트 매트릭스 표현은 규칙트리 구조와 유사하다. 각 규칙의 결론과 모든 규칙들의 전제부를 비교하여 연결이 발견되면, 1 값을 주고, 그렇지 않으면

0 값을 준다. 즉, 규칙의 수가 10개이면 10×10, 100개이면 100×100개의 저장장소가 필요하게 된다.

이 방법을 사용하면 병렬 처리가 가능하여 처리속도를 증진시키고, 규칙의 추가, 삭제 및 수정이 용이하고 추론 경로를 설명하는데 이용할 수 있으며, 의사결정 트리의 어디에서나 추론이 시작될 수 있으며, 다중 결론을 상세화할 수 있다. 또한 환형 규칙(circular rule)의 타당성을 검토하는 알고리즘을 사용하여 검증할 수 있다는 장점을 갖는다.

저장공간의 효율성에 대한 문제점을 해결하기 위해서, 매트릭스를 이중 연결리스트로 변환시키면 컴퓨터 기억공간을 절약할 수는 있으나, 탐색시간의 증가를 가져 온다. 그림 3은 지식베이스를 규칙 비트 매트릭스로 구축한 예를 보여준다. 그러나, 규칙 비트 매트릭스는 절을 구성하는 각각의 fact 정보를 제공하지 못한다는 단점을 갖는다.

지식베이스	
R1 If A Then B AND C	$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$
R2 If B Then D	
R3 If C Then E	
R4 If E OR G Then F	
R5 If J Then I	
R6 If I Then G AND H	
R7 If H Then K	

(그림 3) 지식베이스 예와 규칙 비트 매트릭스의 예  
(Fig. 3) Example knowledge-based and rule bit matrix

(2) 절 비트 매트릭스

규칙 비트 매트릭스의 메모리 손실의 단점을 해결하기 위한 방법이다[19]. 이는 지식베이스에서 절들간의 관계를 보여주기 위하여 소개되었으며, 다음과 같은 순서에 의해서 구현된다.

1. 지식베이스에 있는 규칙의 전제부의 각 절에 고유번호(ID: Identification number)를 할당한다. 예를 들면, 그림 3의 지식베이스에서 ID(4, 2)는 규칙 4의 전제부에 G를 나타낸다는 의미를 갖게 된다.
2. 지식베이스에 규칙들의 결론절의 각 절에 고유번호를 할당한다.
3. 주어진 고유번호를 이용하여 매트릭스를 구성한다.

이와 같이 규칙의 전제부와 결론부를 연결 해 줌으로써 규칙 비트 매트릭스에서 발생할 수 있는 충돌을 방지할 수 있다. 그림 4는 그림 3의 지식베이스를 절 비트 매트릭스로 구성하였다.

(3) 축소 절 비트 매트릭스

절 비트 매트릭스 방법의 단점은 규칙의 수가 많아지면, 트리 형태의 Rete나 TREAT 등의 방법에 비해, 기억공간의 손실이 더 크다는 점이다.

이러한 단점을 극복하기 위한 방안으로 소개된 방법이 축소 CBM[19]이다. 축소 CBM은 CBM을 구축하는 방법과 같이 각 절에 대해 비교하여 매트릭스를 표현하되, 비교 시에 한 가지 조건을 검토한다. 조건이란, "완전한 부정값(0 bit)을 갖는 절인가"의 불리언 값으로, 조건의 값이 긍정일 경우, 계속해서 검토를 하여 매트릭스를 생성시키고, 조건의 값이 부정일 경우, 해당 규칙의 절에서 해당 열과 행을 삭제시킨다.

		전 제 부							
		1	2	3	4	4	5	6	7
결 론 부	1	1	0	1	0	0	0	0	0
	2	1	0	0	1	0	0	0	0
	3	1	0	0	0	1	0	0	0
	4	1	0	0	0	0	0	0	0
	5	1	0	0	0	0	0	0	1
	6	1	0	0	0	0	1	0	0
	7	1	0	0	0	0	0	0	1

(그림 4) 절-비트 매트릭스(CBM)의 예  
(Fig. 4) Example of clause bit matrix

		전 제 부						
		2	3	4	4	6	7	
결 론 부	1	1	0	0	0	0	0	
	2	0	1	0	0	0	0	
	3	0	0	1	0	0	0	
	5	0	0	0	0	1	0	
	6	0	0	0	1	0	0	
	6	0	0	0	0	0	1	

(그림 5) Reduced CBM의 예  
(Fig. 5) Example of reduced clause bit matrix

이는 불필요한 정보를 삭제 시킴으로써 기억공간의 손실을 줄이고자 하는데 목적이 있다. 그림 5는 축소 CBM의 구축 형태를 보여준다.

2.1.3 관계 리스트 표현

생성규칙의 구현방법 중의 하나인 관계 리스트는 특수한 리스트를 사용하여 빠른 탐색과 수정의 용이성을 제공해주며, 최근에는 퍼지 개념을 도입할 수 있다는 장점도 제기되고 있다[19]. 이는 규칙의 결론부에 대해 각기 연계관계를 갖고 있는 규칙의 전제부에 대한 정보를 갖도록 하여 탐색 시에 곧장 찾아 들어갈 수 있게 한다. 관계 리스트는 표 1과 같이 구성된다.

<표 1> 관계리스트 구축 예  
<Table 1> Example of relational list

결 론 절		전 제 절	
규칙번호	절번호	규칙번호	절번호
1	1	2	1
1	2	3	1
3	1	4	1
5	1	6	11
6	1	4	2
6	2	7	1

2.2 추론 방법

추론방법에는 전향추론, 후향추론, 직접추론, 혼합추론 방식 등이 있다. 전향추론은, 일련의 사실들에 의해서 결론을 유추해 나아가는 방식으로 목표가 잘 알려져 있지 못할 경우에 유용하며, 최근 가장 인기 있는 패턴매칭 방법인 Rete 네트워크 구성[5, 20]으로 인해 많이 연구되어온 운영 방식이다. 후향추론은, 목표 설정 후 적합한 규칙을 찾아 증명하여 처음 설정되었던 목표가 정확한지를 밝혀주는 방식으로, 목표가 잘 알려져 있을 경우 유용하며, 시스템은 그 목표에 맞는 증거를 찾는 역할을 한다. 직접추론은, 비트 매트릭스 또는 관계 리스트에 의한 지식표현 기법으로 구축되어진 지식베이스를 이용하며, 어디에서나 추론이 시작될 수 있으며, 가능한 많은 결론에 도달

할 수 있는 장점과 구현이 매우 간단하고 속도가 빠르다는 장점을 가지고 있다. 끝으로, 이들 세 가지 방법들을 혼합하여 사용할 수 있도록 하는 혼합추론이 있다. 혼합추론의 경우 사실들을 조합하여 근접한 목표를 찾거나, Meta 규칙을 검색하는 과정은 전향추론으로, 가설을 통해 검증하는 단계를 후향추론으로 이용을 한다[18].

이러한 추론운영 방식은 지식베이스 시스템 내에서 독립적인 구성원으로 존재하지 않으며, 적용되는 도메인의 지식표현 기법에 따라 선택될 수 있다. 그 예로 Rete, TREAT 등과 같은 패턴 네트워크를 이용한 추론엔진의 경우에 있어서는 대부분 전향추론 방식을 택하고 있다[5, 15, 17, 22].

### 3. 새로운 지식구현 방법

#### 3.1 기존 지식구현 방법들의 문제점

추론을 효율적으로 하기 위해서 이용되는 트리 표현은 탐색이 선형적이기 때문에 매우 느리고, 지식베이스의 타당성 검증이 어려우며, 지식표현을 사용하기에 비실용적인 경우가 많다는 단점을 갖고 있다. 트리 표현의 변형으로, 다수의 객체와 다수의 패턴매칭을 하는데 유용하게 사용되며, 전향추론 시에 효율적인 Rete는 알파메모리, 베타메모리라는 저장공간이 추가로 요구되며, 규칙이 변경되면 전체를 재구성해야하며, 규칙이 적을 경우 효율적이지 못하다는 단점을 갖고 있다. 이러한 Rete의 저장공간 효율성의 문제를 해결하기 위해 제안된 TREAT 역시 탐색속도의 문제점을 갖고 있다. 그러므로 추론속도가 빠른 추론엔진 구성을 위해서는 이 표현기법이 적절하지 못하다[8].

규칙 비트 매트릭스는 절들간의 관계에 대한 정보를 제공하여 주지 못하고, 대부분의 불필요한 정보를 저장해야 하는 단점을 갖고 있으며, 이 점을 보완한 절 비트 매트릭스는 규칙에 포함되어 있는 AND/OR 조건을 실행시킬 수 있는 표현이 없어 추론 시에 AND/OR를 고려한 추론이 어렵다는 단점을 갖는다.

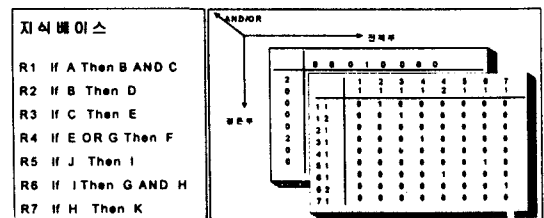
절 비트 매트릭스에 대한 효율적인 메모리 관리를 위한 연구의 결과로, 최근 발표된 축소 절 비트 매트릭스는 불필요한 널값을 보관하는 부분을 제거하는 방법이다. 그러나, 절 비트 매트릭스나 축소 절 비트

매트릭스 모두, 규칙에 고유 조건이 존재하는 것을 무시하고 직접 연결되어진 규칙들만을 찾아 나아간다면, 결코 바람직한 해를 도출해 낼 수 없다는 단점을 갖고 있다[18]. "If A and B Then C"라는 규칙이 있을 경우, 전제부의 두 값 중 하나의 값만 검증되어 참을 나타낸다면 다른 한 값의 검증여부를 검토하지 않고, 결론부인 "C"를 활성화 시키는 불완전한 결과를 보일 수 있다는 것이다. 또 다른 예로, "and" 조건을 만족하는 두가지의 결론에 대해 조건 탐색을 하지 않은 경우, 불완전한 값을 보일 수 있다.

이 논문에서는 추론의 속도가 빠르고, 대다수의 도메인에 적용이 용이하고 규칙의 검증이 가능한 절 비트 매트릭스 방법과 AND/OR의 조건 연산이 가능한 트리구조의 장점을 조합한 3차원 지식구현 방법을 제안한다.

#### 3.2 ECBM 방법의 제안

ECBM은 2차원 테이블인 CBM에 AND/OR 상태 표시 테이블을 추가하여 3차원으로 구성함으로써 추론 시에 규칙의 조건을 고려하여 완전한 추론을 할 수 있도록 해 준다. 즉, 기존의 CBM에 AND/OR연산을 가능하게 하기 위해 조건상태 표시를 할 수 있는 비트를 추가하여 그림 6와 같은 3차원 구조를 갖도록 설계하였다.



(그림 6) ECBM의 구축 형태  
(Fig. 6) Construction type of ECBM

예를 들면, CBM을 이용하여 추론을 할 경우에는 규칙이 "If A AND B Then C"라 했을 때, 전제부의 "AND"에 대한 별도의 조치가 없이 먼저 읽어들이는 A에 대한 탐색, 그 후 B에 대한 탐색을 해 나아간다. 그러나, 여기에서 ECBM은 규칙에서 AND를 인식하여 저장하고 추론 시 이를 이용하여 AND조건이 모

두 만족할 때 추론이 계속될 수 있도록 한다.

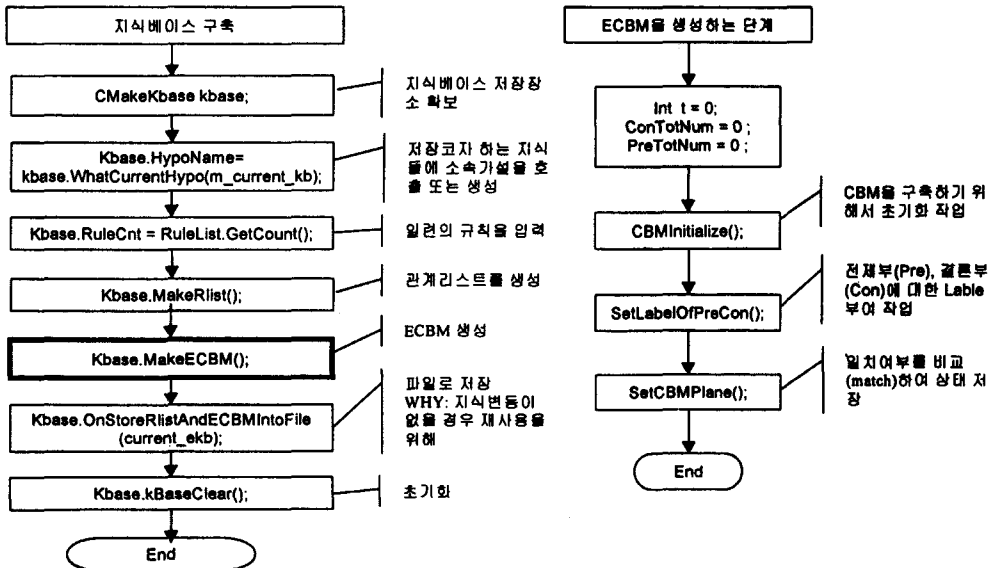
ECBM의 장점은, 첫째로, ECBM은 일단 생성된 후, 이를 이용할 수 있는 추론운영 방식에 제한을 두지 않는다. 이는 연구의 주요목적인 범용성을 갖는 추론엔진을 가능하게 해 주는 중요 장점이다. 둘째, 지식의 검증이 용이하다. GIGO(garbage in garbage out)의 원리에 따라 잘못된 정보를 갖고 추론을 한 결과의 정확성은 당연히 의심을 받게 된다. 이러한 부정확한 결과를 산출하지 않기 위해 지식의 검증이 반드시 필요하지만 트리 표현 등 기타 방법을 이용한 지식베이스 구축에서는 검증이 쉽지 않다. 셋째, 특정 도메인에 국한되지 않아 범용성을 갖는다는 장점을 갖고 있다. 반면에 ECBM을 지식구축 방법으로 이용할 경우 규칙의 수가 많아질수록 그에 비례하여 많은 널값을 저장해야 하므로, 이에 따른 메모리의 손실을 감수해야 한다는 단점을 가지고 있다.

그림 7은 이 논문에서 제안한 ECBM을 구축하는 중요 모듈인 CBM 구축을 위한 초기화 작업을 하는 *CBMInitialize* 모듈, 전제부와 결론부에 대한 고유 인 식 번호를 주는 작업 모듈인 *SetLabelOfPreCon*, 매트

릭스에 값을 부여하는 작업을 하는 *SetCBMPlane* 모듈을 보여주고 있다.

### 3.3 ECBM을 이용한 추론엔진

추론엔진 중에서 가장 비중을 많이 차지하는 것은 패턴매칭(pattern matching)을 수행하는 매치이다. 어떠한 알고리즘을 적용하면 효율적인지에 대한 많은 연구가 이루어 졌으며, 그 결과 *Rete*, *TREAT* 등이 발표되어졌다. 다수의 객체와 다수의 패턴매칭을 하는데 유용하게 사용되며 전향추론 경우에 효율적인 *Rete*는 알파메모리, 베타메모리라는 저장공간이 추가로 요구되면 규칙이 변경되면 전체를 재구성 해야하며, 규칙이 적을 경우 효율적이지 못하다는 단점을 갖고 있다. 이러한 *Rete*의 저장 공간 효율성의 문제를 해결하기 위해 제안된 *TREAT* 역시 탐색속도의 문제점을 갖고 있다[8]. 또한 전향 추론이라는 제한된 운영 방식을 택하고 있어 이 논문의 목적인 범용적인 추론을 위한 매칭 알고리즘으로는 적합하지 못하다. 따라서 이 논문에서는 ECBM에 맞는 매치를 별도로 구현하는 방법을 택하였다.



(그림 7) 지식베이스 구축과 ECBM 구축 흐름도  
(Fig. 7) Construction of knowledgebase & flowchart of ECBM construction

추론과정을 설명하기 전에 사전 정의적인 요소들은 다음과 같다.

첫째, 블랙보드는 단순 작업메모리 기능을 수행하며, 사용자나 컴퓨터 검토에 의해서 입력된 초기치를 갖는다.

둘째, 추론방법은 사용자의 선택에 의해 결정된다.

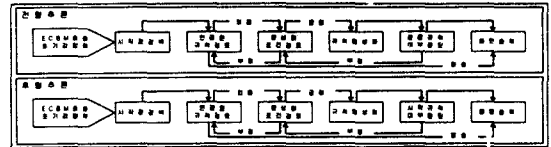
셋째, 추론 함수의 호출 시에 사용되는 값은 블랙보드의 초기치와 지식베이스이다.

넷째, 충돌전략은 선입선출(FIFO)을 기본값으로 설정한다.

다섯째, 직접추론은 가설 추론과 메타 규칙에 대해 적용되어지는 추론 운영 방식이다.

ECBM을 이용한 전향추론과 후향추론의 수행순서는 그림 8과 같다. 그림 8에서 보여주듯이, ECBM을 이용한 추론은 매트릭스의 형태를 기본으로 하고 있기 때문에 추론방식이 다르더라도 수행에 기본 절

차는 같다. 다만 전향추론일 경우, 연결된 규칙 검색 부분과 규칙 검토부분에서 백트래킹(backtracking)과 충돌문제 해결(conflict resolution)등의 다소 복잡한 알고리즘을 추가적으로 사용한다는 점이 다르다.



(그림 8) ECBM을 이용한 전향·후향추론의 흐름도  
(Fig. 8) Flowchart of forward and backward chaining of using ECBM

표 2는 ECBM을 이용한 전·후향추론의 단계별 비교를 나타낸다.

〈표 2〉 ECBM을 이용한 전향·후향추론의 단계별 비교  
(Table 2) Comparison of forward and backward chaining of using ECBM

단계	전향추론	후향추론
1 단계	ECBM에서 초기 증상값과 맞는 StartPoint를 찾음 ① StartRule의 전제부가 블랙보드에 있는가?를 판단 ㉠ 긍정이면, SelectRule = StartRule ㉡ 부정이면, BackTracking을 시도 긍정이면 ㉠을, 부정이면 1단계로. ② ①의 실행결과 일치값이 없으면, 추론실패출력 후, 종료 ③ AND/OR조건 검토	ECBM에서 초기값과 맞는 StartPoint를 찾음 ① StartRule의 결론부가 블랙보드에 있는가?를 판단 ㉠ 긍정이면, SelectRule = StartRule ㉡ 부정이면, BackTracking을 시도 긍정이면 ㉠을, 부정이면 1단계로. ② ①의 실행결과 일치값이 없으면, 추론실패출력 후, 종료
2 단계	블랙보드에 SelectRule의 결론부를 배치	블랙보드에 SelectRule의 전제부를 배치
3 단계	SelectRule의 결론부를 전제부로 갖는 Rule을 찾아 ConflictSet에 저장	
4 단계	충돌전략을 이용해 우선순위가 높은 Rule을 선택하여 SuspendRule로 상태 변화를 시킴	SelectRule의 전제부(들)을 결론부로 갖는 Rule을 ECBM에서 탐색
5 단계	SuspendRule에 다른 전제부가 있는가를 확인 ① 긍정이면, 블랙보드와 전제부를 비교 ㉠ 일치하면, 다음으로 계속. ㉡ 일치하지 않으면, BackTracking을 시도 긍정이면, 다음으로 계속. 부정이면, SuspendRule = Null, 4단계로. ② AND/OR조건 검토	① AND/OR조건탐색 ② 전제부가 여럿일 경우, 먼저 들어온 전제부에 대응하는 규칙은 FiredRule로, 나머지는 SuspendRule로 상태 변환 ③ 블랙보드에 값이 있는가? ㉠ 긍정 : 5단계로 ㉡ 부정 : 4단계로
6 단계	5단계의 ②에 결과값이, ① 긍정이면, SuspendRule을 FiredRule로 상태변환 ② 부정이면, 4단계로.	



7 단계	FiredRule의 결론부를 블랙보드에 추가	FiredRule의 전제부를 블랙보드에 추가
8 단계	ConcludingRule인가를 확인 ① 긍정이면, 결과출력후 9단계로. ② 부정이면, SelectRule에 FiredRule를 대치, 3단계로.	InitialRule인가를 확인 ① 긍정이면, 결과출력후 6단계로. ② 부정이면, SelectRule에 FiredRule를 대치, 3단계로.
9 단계	다른 StartingRule이 있는가?(다른 초기증상이 있는가?) 확인 ① 긍정이면, 1단계로. ② 부정이면, 끝	다른 StartRule로 추론을 계속할 것인가? 확인 ① 긍정이면, 1단계로. ② 부정이면, 끝.

#### 4. 평 가

이 연구를 통해 구축하려는 추론엔진에 대한 요구사항을 살펴보면, 첫째이 연구를 통해 구축하려는 추론엔진에 대한 요구사항을 살펴보면, 첫째, 혼합추론이 가능한 엔진이므로, 추론운영 방식에 제한이 없어야 한다. 이는 전향추론 및 기타 추론운영 방식에 있어서 변화에 따른 성능 저하가 없어야 한다는 조건을 만족해야 한다. 둘째, 추론의 결과출력이 빨라야 하며, 정확해야 한다는 것이다. 이러한 요구사항에 적합하게 구성되어 졌는가에 대해서 살펴보고, Rete 네트워크를 이용하는 일반적인 추론엔진과 비교를 하였다.

이 논문에서는 추론엔진을 설계 및 구현시에 패턴 매처로 Rete를 선택하지 않았다. 그 이유는 Rete 네트워크를 이용한 추론엔진이 후향추론이나 직접추론 등에는 이용하기가 어렵거나, 이용하였을 경우에 그 성능이 현저하게 떨어지는 특성이 있어 대부분의 연구에서 전향추론 이외의 추론 운영 방식에는 적용하지 않아왔다는 점이 범용적인 성격을 갖는 엔진을 구축하고자 하는 목적에적합하지 않았기 때문이었다.

패턴 매처가 Rete 알고리즘을 사용할 경우 생성되어지는 Rete망과 ECBM의 특성을 간단히 살펴보면 표 3과 같다. ECBM은 매트릭스 구조로 지식베이스로 생성규칙을 적용할 수 있는 특성을 가진 실세계의 도메인이라면 대부분 수용할 수 있는 일반성(generalization)을 가지고 있으나, Rete는 트리 구조 형식으로 트리 구조를 이룰 수 있는 구조, 즉 트리 구조를 이룰 경우 효율적인 도메인 형태에만 적용할 수 있는 특수성(specialization)을 갖는 메카니즘이다.

이 연구의 추론에 이용된 지식베이스 ECBM은 규칙의 입력과 동시에 생성되도록 구성되었으며, 생성된 ECBM을 이용하여 같이 흐름에 따라 표 2와 같

이 전향추론과 후향추론을 실행한다. 또한 ECBM 생성과정의 중간단계로 관계 리스트가 구축되며, 이를 이용하여 직접추론을 수행하도록 설계하여 범용적인 추론엔진 구축의 요구사항을 만족하였다.

〈표 3〉 Rete와 ECBM의 특성비교  
(Table 3) Properties comparison of rete with ECBM

항목	ECBM	Rete
방식	비트 매트릭스 구조	트리 구조
장점	추론운영방식에 제한이 없음 지식베이스의 검증이 용이 및 운영 용이 추론 속도가 빠름 특정도메인에 국한되지 않아 범용성을 갖음 AND/OR 연산수행	다수의 객체와 다수의 패턴 매칭에 효율적임 전향추론에 효율적임 규칙의 삽입 삭제 용이
단점	규칙의 수가 많을 경우, 분할 기법이 필요 메모리의 효율성이 낮음	메모리의 효율성이 낮음 Rete망 구성이 어려운 도메인들이 많음 추론운영방식에 제한이 있음
기타	이 연구에서 제안한 구조	기존 연구에 의해 검증된 구조

표 4는 ECBM을 이용한 추론엔진을 실험하기 위해 주어진 도메인으로부터 지식베이스를 구축하기 위해 규칙들을 입력하는 형태를 보여주고 있다. 여기서 사용된 규칙은 삼성 중공업의 수많은 도메인 중 13개의 센서를 통해 들어오는 자료를 받아 진단하는 도메인을 현장 작업자에게 제공받아 표 4와 같은 형식에 맞추어 정형화(formalization)를 시켜 규칙을 생성하였다.

〈표 4〉 사용된 규칙의 예  
 〈Table 4〉 Example of using rule

Rule-name: R1	Rule-name: R2
Rule-class: Domain Rule	Rule-class: Domain Rule
Rule-id: 1	Rule-id: 2
IF:	IF:
Tokens: <i>Sensor1.Q is TRUE</i>	Tokens: <i>Sensor1.M is TRUE</i>
Speech: float-var is true	Speech: float-var is true
CF: 0.000000	CF: 0.000000
THEN:	THEN:
Tokens: <i>Sensor1.X = TRUE</i>	Tokens: <i>Sensor1.Y = TRUE</i>
Speech: float-var = true	Speech: float-var = true
CF: 0.000000	CF: 0.000000
Rule-priority: 0	Rule-priority: 0

그림 9는 전향추론의 경우에 트리구조로 Rete 매칭 알고리즘을 매치의 기법으로 선택한 일반적인 추론엔진과 ECBM을 이용하여 구현한 추론엔진과의 속도의 변화에 대한 그래프이다. Rete의 특성 상, 추론 운영 방식 중에서는 전향 추론만 비교를 하였다. 그림 9(a)는 지식 구축을 하는데 걸리는 시간을 검토하여 비교한 그래프로 트리구조로 구축할 경우 보다 좀더 효율성이 좋은 것을 보여주고 있다. 자료구조를 매트릭스로 사용할 경우 즉, ECBM으로 구축할 경우 구축하는데 걸리는 시간은  $O(n)$ 이며, 트리 구조로 저장할 경우 즉 구축할 경우는  $O(n \log_2 n)$ 이라는 값을 갖

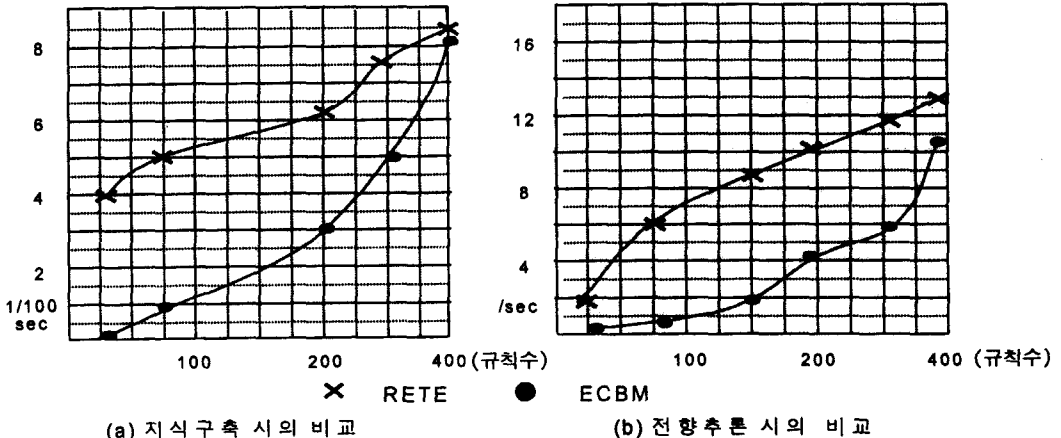
게 된다.

전향추론의 경우 비교인 그림 9(b)는 기존에 트리 운영으로 효율성을 갖았던 Rete를 이용한 패턴 매칭 알고리즘을 적용한 추론엔진과 비교하여 성능이 좋을 것을 보여 주고 있다. 또한 전향 추론 뿐만 아니라, 후향 추론·직접추론 등을 할 수 있는 융통성을 지닌 것이 장점이나 메모리를 많이 차지한다는 점과 아직 많은 검토를 거친 검증된 구조가 아니라는 단점이 있다.

### 5. 결 론

전문가 시스템을 구현하기 위해서는, 적용되어질 도메인의 특성을 파악한 후에, 구체적인 구현을 하는 것이 실패하지 않는 지름길인 것이다. 전문가 시스템 중 추론엔진에서 추론방법은 무엇을 택할 것인가, 추론시에 이용될 지식표현은 어떻게 구성할 것인가에 대한 질의와 답변이 행해져야 할 것이며, 이후 지식 표현을 어떠한 방법으로 구축하였을 경우에 추론의 효율성을 높여 줄 것인가를 결정하여, 구체적인 시스템을 구축해야 한다.

이 논문에서는 추론의 기본이 되어지는 지식을 지식베이스로 구축하는 방법에서의 개선과 이에 적합한 매치를 설계하는 것을 통해 추론엔진과 지식베이스의 양측면에서의 성능향상을 도모하는 것이 목적으



(그림 9) Rete와 ECBM을 이용한 추론엔진 성능 비교  
 (Fig. 9) Efficiency test of inference engine of using rete and ECBM

로, 범용성을 갖는 전문가 시스템 셸을 구축하는 주요 구성 요소 중에서 지식베이스 구축형태인 ECBM를 제안하여 이에 맞는 매처를 갖는 추론엔진을 구축하였다.

추론에 이용되는 지식의 형태 즉, 지식베이스 구축을 위한 표현형태는 표준화와 모듈화가 용이한 성격이 있는 생성규칙을 선택하였다. 이러한 성격은 사용자가 지식을 습득하고 사용하는데 있어 편리성을 제공하여 준다. 지식베이스 구축의 방법은 트리구조가 아닌 비트 매트릭스를 근간으로 하여 제안한 ECBM로 선택하였다. 이는 구축 속도가 트리 구조보다 빠르며, 대부분의 도메인에 적용이 가능하다는 보편성을 갖고 있다. 제안한 ECBM을 이용하여 전향추론, 후향추론, 직접추론을 선택하여 수행할 수 있다는 장점도 추가적으로 확보할 수 있었다.

전향 추론인 경우에는 Rete 매칭 알고리즘을 이용하는 매처와 비교할 수 있었지만, 그 외에 후향추론과 직접추론에 대한 비교는 Rete 매칭 알고리즘을 이용한 매처는 이 두 추론방식을 수용할 수 없기 때문에 이루어 지지 않았다. 이러한 점 역시 이 논문에서 주장하는 ECBM을 이용한 추론엔진의 장점 중에 하나가 된다.

앞으로, 제안된 ECBM과 구현된 추론엔진은 기본적인 기능만을 수행하여 준다. 규칙을 지식획득과 지식베이스를 구성하는 표현으로 선택하였기 때문에 발생하는 문제점을 위해 해결책으로 지식을 검증하여 획득하고 구축하는 지식 획득 모듈이나 자기 학습(learning) 모듈 등의 추가 등의 개발이 요구된다. 또한 최근에 관심에 집중이 되고 있는 분야 중에 하나가 바로 실시간 인공지능 시스템(real time artificial intelligence system)이다. 이 논문은 실시간 개념을 도입하여 구현하지는 않았지만 실시간 개념을 도입할 경우 확장성의 용이성을 고려하여 설계·구현되었다. 향후 연구 과제로 이 논문에서 구현한 추론엔진에 실시간 개념을 도입하여 실시간 시스템으로 확장을 하는 연구가 필요하다.

## 참 고 문 헌

- [1] Barr, A. and E. A. Feigenbaum, "The Handbook of Artificial Intelligence," Vol. I, William Kaufman, 1981.
- [2] Bourbakis, Nikolaos G., "Knowledge Engineering shells: systems and techniques," World scientific publishing Co. Pte. Ltd. 1993.
- [3] Buchanan, Bruce G., Edward H. Shortliffe, "RULE-BASED EXPERT SYSTEMS: The MYCIN experiments of the stanford heuristic programming project," Addison-Wesley Publishing Company, 1985.
- [4] Carlis, John V., James P. Held, "Conceptual Data modeling of Expert System," IEEE EXPERT, Spring 1989.
- [5] Forgy, C. L., "Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem," Artificial Intelligence 19: 17-37, 1982.
- [6] Giarratano, Joseph, Gary Riley, "Expert System: principles and programming," PWS publishing company, 1994.
- [7] Ginsberg, Matthew L., "Dynamic Back tracking," Journal of Artificial Intelligence Research 1 25-46, 1993.
- [8] Hanson, Eric N., Yu-Wand Wang, "A Performance Comparison of the Rete and TREAT Algorithms for Testing Database Rule Conditions," IEEE, 1992.
- [9] Hanson, Eric N., "Gator: A Discrimination Network Structure for Active Database Rule Condition Matching," April 7, 1993.
- [10] Hayes-roth, Frederick, "RULE-BASED SYSTEMS," CACM Vol. 28, No. 9, September 1985.
- [11] Hopgood, Adrian A., "KNOWLEGE-BASED SYSTEMS for Engineers and Scientists," CRC press, 1993.
- [12] Jackson, P., "Introduction to Expert Systems," Addison-Wesley, 1990.
- [13] Keravnou, E. T., Johnson, L., "Competent Expert Systems: A case study in falut diagnosis," Kogan Page Ltd, 1986.
- [14] Mettrey, William, "An Assessment of Tools for Building Large Knowledge-based Systems," AI

magazine, winter 1987.

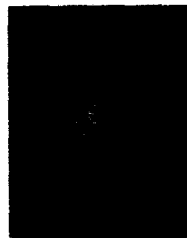
- [15] Miranker, Daniel P., "TREAT: A better match algorithm for AI production systems," In Proc. AAAI National Conference on Artificial Intelligence, 42-47, August 1987.
- [16] Rich, E., "Artificial Intelligence," McGraw- Hill, pp. 180-183, 1983.
- [17] Scales, Daniel J., "Efficient Matching Algorithms for the SOAR/OPSS production system," STAN-CS-86-1124, June 1986.
- [18] Schmoldt, Daniel L., H. Michael Rauscher, "Building Knowledge-based systems for Natural Resource management," 1996, ITP.
- [19] Schneider, Moti, Abranham Kandel, Gideon Langholz, Gerard Chew, "Fuzzy Expert System Tools," Bookcraft(Bath) Ltd., 1996.
- [20] Stefik, Mark, "Introduction to Knowledge Systems," Morgan Kaufmann Publishers, Inc., 1995.
- [21] Stephen L. Scott, "Optimal pattern distributions in Rete-based production systems," Third Conference on CLIPS Proceedings, September 1994.
- [22] Tambe, Milind, Dirk Kalp and Paul Rosenbloom, "Uni-Rete: Specializing the Rete Match Algorithm for the Unique-attribute Representation," CMU-CS-91-180, September 1991.
- [23] 류근호, "전문가 시스템 셸 연구보고서," 충북대학교, 1997년 7월.
- [24] 오명륜, 전근환, 허경, 이양원, 류근호, 도기상, "실시간 고장 진단 전문가 시스템 셸을 위한 추론엔진 설계," 한국정보처리학회 추계 학술발표논문집 제3권 제 2호, 1996.
- [25] 오명륜, 오광진, 류근호, 이양원, 김영훈, "ECBM을 이용한 추론엔진 설계," '97 정보과학회 봄 학술발표 논문집, 24권 1호, 1997. 4.
- [26] 이재규의 5명 공저, "전문가 시스템 # 1: 원리와 개발," 법영사, 1996. 10.



**신 정 훈**

1982년 숭실대학교 전자계산학과 졸업  
 1991년 충북대학교 대학원 전산통계학과(이학석사)  
 1995년 충북대학교 대학원 전자계산학과 박사과정 수료  
 1992년~현재 이리농공전문대학 전자계산과 조교수

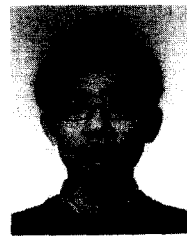
관심분야: 시간지원 데이터베이스, 지식베이스 시스템, 운영체제



**오 명 륜**

1993년 충북대학교 경영정보학과 졸업  
 1996년~현재 충북대학교 대학원 전자계산학과 석사과정

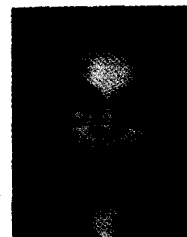
관심분야: 시간지원 데이터베이스, 지식베이스 시스템



**오 광 진**

1996년 충북대학교 컴퓨터학과 졸업  
 1996년~현재 충북대학교 대학원 전자계산학과 석사과정

관심분야: 시간지원 데이터베이스, 지식베이스 시스템



**이 양 원**

1979년 숭실대학교 전자계산학과 졸업  
 1983년 연세대학교 산업대학원 전산전공(공학석사)  
 1994년 숭실대학교 대학원 전산전공(공학박사)

1979년~1986년 국방연구원 근무  
 1986년~현재 군산대학교 컴퓨터학과 교수 겸 정보통신 기술 연구소장

관심분야: 영상 데이터베이스, 지식베이스 시스템, 컴퓨터 비전



**류 근 호**

- 1976년 숭실대학교 전산학과 졸업(학사)
- 1980년 연세대학교 산업대학원 전산전공(공학석사)
- 1988년 연세대학교 대학원 전산전공(공학박사)

1976년~1986년 육군군수 지원사

전산실(ROTC 장교), 한국전자통신연구소(연구원), 한국방송통신대 전산학과(조교수) 근무

1989년~1991년 Univ. of Arizona Research Staff (TempIS 연구원, Temporal DB)

1986년~현재 충북대학교 컴퓨터과학과 교수겸 컴퓨터 정보통신 연구소장

관심분야: 시간지원 데이터베이스, 시공간 데이터베이스, DBMS 및 OS, 객체 및 지식베이스 시스템



**김 영 훈**

- 1986년 한국항공대학교 전자공학과 졸업
- 1988년 동국대학교 대학원 전자공학과
- 현재 삼성중공업(주) 중앙연구소 메카트로 개발센터 선임연구원

관심분야: 전문가 시스템 셀 개발, 지식 기반 시스템 구축 및 공정 자동화 시스템 등의 응용 시스템 개발