

실시간 전문가 시스템의 개발 및 활용에 관한 연구

황하진¹⁾

<Abstract>

As the application of real time expert systems has been expanded, the real time expert systems have become an integral part of the information society. Real time expert systems are viewed as a new paradigm to integrate real time problem solving and expert system technology. However, a substantial amount of work is still required to effectively handle a new and challenging opportunity for successful implementation of real time expert systems.

In this article, the basic concepts and characteristics of real time expert systems are discussed. The article also presents, through the literature survey, the summary of real time expert systems applications and a class of tools, providing an enabling real time based expert system technology. Finally, the article discusses a set of requirements which real time expert systems must satisfy.

I. 서론

오늘날 고도로 발달한 산업사회에서 정보시스템의 도입이 확산되고 경쟁환경이 극도로 첨예하게 변화함에 따라, 신속한 업무처리와 처리결과에 대한 중요성이 날로 높아지고 있다. 실시간 전문가 시스템(Real Time Expert Systems)은 특정업무처리가 주어진 시간내에 처리하지 못할 경우 시스템의 존재가 무의미 해지거나, 시스템의 성능에 결정적인 영향을 미치는 경우에 적용하기 위한 정보시스템의 새로운 접근방법으로써 최근들어 이에 대한 활발한 연구가 진행되고 있다. 실시간전문가시스템 운영의 기본적인 토대는, 산업의 고도화와 업무의 구조적 복잡도가 증가함에 따라 보유자원과 인간의 대응능력의 한계를 극복 하고자 실시간시스템과 전문가시스템기술을 접목 시킴으로써 시너지 효과를 극대화하려는 시도라고 할 수 있다.

전문가 시스템은 인공지능의 여러 연구분야중 대표적으로 성공적인 응용분야의 하나로

1) 대구효성가톨릭대학교 경상대학 조교수

본 연구는 대구효성가톨릭 대학교 1996년 교비 연구비의 지원에 의한 것임.

써, 인간 전문가의 문제해결 능력을 컴퓨터가 갖게 하는 것이다. 전문가 시스템에 관한 연구는 지식의 표현 및 습득방법, 추론제어방법 등이 활발하게 개발되어 각종 응용분야에서 다양하게 활용되고 있으며, 응용분야의 특성에 적합한 다수의 전문가 시스템이 개발되어 왔다. 그 중에서 실시간 전문가 시스템은 환경제한요소 중 시간적인 요소가 가장 중요시 되는 시스템으로서 특수한 분야에 응용되어 왔으나 전문 인력자원의 제한을 극복하기 위해 점차 그 활용의 중요성이 더해 가고 있다.

특히, 전문가시스템의 활용이 예술분야에서 부터 공학분야에 이르기까지 확대되면서 더욱 의욕적으로 접근할 수 있는 활용분야의 개척이 필요하게 되었다. 오늘날 실시간 처리 시스템은 우리들의 일상 생활에서 중요한 자리를 차지하고 있다. Wright등[1986]에 의하면 간단하고 규모가 작은 가정용품에서 부터 복잡하고 대규모의 산업 및 군사적 목적의 실시간 처리 시스템의 개발과 응용이 증가 추세에 있다. 이와 같은 연구노력 속에서, 미개척 분야의 하나로서 가장 관심있고 매력적인 분야가 바로 실시간 영역에의 전문가 시스템의 활용이다. 공중경보 시스템이나 위기관리 시스템 등과 같이 실시간 상황에서 운용되는 전문가 시스템은 특별히 비동기적 사건의 흐름과 다이내믹하게 변화하는 시간적 제약요인 및 하드웨어 등의 요구사항에 민감하게 대처해야 한다. 따라서, 실시간 시스템 환경에서는 시간적 추론 (Temporal Reasoning), 비 단조성 (Non-monotonicity), 인터럽트 조절 기능 및 불완전한 입력 데이터의 조작 방법 등을 수용하면서 엄격한 시간적 제약 요인을 충족시킬 수 있는 추론능력을 갖춘 유연성 있는 소프트웨어 구조가 요구된다. 한편, 실시간 처리 시스템의 기능적 복잡도는 크게 3가지 측면에서 급격히 증가하고 있는데 이를 살펴보면[Laffey, 1988; Hadavi, 1992],

첫째, 통제 해야할 기능의 수

둘째, 통제되는 기능의 비율

셋째, 어떤 판단 또는 결정이 내려지기 전에 고려되어야 할 주요 요인들(Factors)의 수이다.

이와 같은 기능적 복잡도의 증가는 실시간 처리 시스템과 전문가시스템 기술의 접목에 대한 상당한 관심을 불러 일으켰다. 실시간 전문가 시스템은 계산결과가 이론적으로 정확하여야 할 뿐 아니라 결과가 생성되는 시간적 조건을 정확하게 만족하도록 작업을 수행해야 한다. 즉, 실시간 전문가 시스템은 그 시스템에 요구된 기능을 주어진 시간제한(Time Constraints) 범위 안에서 모두 시행되고 시스템 외부환경과의 상호작용이 정확하고 신뢰성 있게 이루어지도록 제어되어야 한다. 실시간 처리 전문가시스템의 개발에 대한 연구는 실시간 시스템의 영역을 인공위성의 통제, autonomy, 데이터 분석, 자동기관(전동)차, 전투관리, 항공통제 시스템, 컴퓨터 통신망의 통제 및 감시, 로봇트 공학 및 비전 시스템 (Vision Systems), 마케팅 및 재무정보관리 시스템, 그리고 환자감시 등의 의학 분야에 이르기까지로 확대하는 계기가 되었다.

본 논문에서는 확산되고 있는 실시간 전문가 시스템에 대한 관심과 연구 노력에 적절히 대응하기 위하여 첫째, 실시간 전문가 시스템의 기본개념과 특성, 실시간 전문가 시스템 연구와 관련한 주요 문제점, 그리고 미래 연구분야를 살펴보고, 둘째, 문헌연구를 통하여, 실시간 전문가 시스템 개발도구와 그 개발 및 활용 사례를 알아보고, 실시간 전문가 시스템 개발도구 선택시 고려해야 할 사항들을 제시하고자 한다.

II. 실시간 전문가 시스템의 기본개념과 필요성

2.1. 실시간 전문가 시스템의 기본 개념

2.1.1. 정의

“실시간”이라는 말은 정의를 내리기 보다는 그냥 감각적으로 느끼는 것이 훨씬 쉬울 수 있다.

예를 들어, 마그네틱 테이프에 저장된 은행 거래자료를 밤새 수정작업하는 컴퓨터 프로그램을 실시간 시스템이라고 부르지는 않는다. 그러나, 비행기의 자동 항법장치를 조절하는 컴퓨터 프로그램을 실시간 시스템이라고 부르는데는 아무도 주저하지 않을 것이다. “실시간” 시스템에 대해서는 많은 정의가 존재한다. 우선 가장 보편적인 실시간에 대한 정의는 데이터를 신속히 처리 할 수 있는 “신속성”이다. 또 다른 정의는 “인지적 신속성”인데 즉, 사람이 할 수 있는 것보다 빨리 해낼 수 있는 인지적 능력을 의미한다. 이런 측면에서 실시간 시스템의 정의는 “데이터가 입력되는 속도 만큼 신속하거나 또는 더 신속하게 반응하는 시스템”[Mandelkern, 1992; Stoyenko, 1993; Zhu, 1993] 이라고 할 수 있다.

문헌 조사에 의하면, 실시간 시스템에 대한 정의는 다음과 같이 두가지로 요약할 수 있다.

첫째, 처리대상 업무에 대하여 예측가능하게 신속히 실시간적 행위를 구사할 수 있는 시스템이며,

둘째, 어떤 알고리즘이 채택 되는가에 관계없이 시스템의 응답시간에 대한 엄격한 제한이 존재 하는 시스템이다.

응답시간(Response Time)은 컴퓨터가 외부사건을 인지하고 반응하는데 소요하는 시간을 의미한다. 실시간 시스템의 활용에 있어 이 반응시간의 측정이 가장 중요한 요소이다. 왜냐하면, 발생사건들이 엄격한 시간 통제에 의하여 처리되지 않으면 전체 처리대상 프로세스는 통제불능 상태에 빠지기 때문이다. 그러므로 실시간 시스템의 주요 특징으로서 어떤 의사결정 상황에서 “주어진 시간”이 아주 중요한 제약 요인이 될때, 그 “주어진 시간”요건을 충족시켜 줄 수 있는 대응능력을 들 수 있다. 만일 불규칙적인 데이터의 입력이나 시스템의 임의적 상황이 주어졌을때 시스템이 필요한 시간내에 적절한 응답을 할 수 있다면 그 시스템은 ‘실시간’ 시스템이라고 할 수 있다.

실시간 전문가 시스템은 일반적으로 센서를 통해 입력되는 외부의 사건에 반응하고 다중작업의처리를 분할하는 기능들을 가지며 요구들은 주기적 혹은 비주기적으로 처리된다. 실시간 시스템의 특성을 수용한 실시간 전문가 시스템의 요구사항을 요약하면 다음과 같다[Natarajan, 1992; Wesbin, 1987; Ali, 1985].

첫째, 엄격한 응답시간의 충족과 높은 수준의 시스템 수행능력이 요구된다.

둘째, 대부분 CPU, 메모리, 통신 등의 자원제약(resource constraint)을 받는다.

셋째, 비 동기적 사건의 처리능력, 즉, 스케줄에 일치하지 않고 발생하는 사건이나 비동기적 사건들도 효과적으로 다룰수 있어야 한다.

넷째, 시스템의 일부에 오류가 발생하더라도 지속적으로 운영될 수 있는 능력을 갖추어야 한다.

다섯째, 시간적 추론능력과 불완전한 데이터가 입력되더라도 적절히 처리 할 수 있는 능력을 갖추어야 한다.

2.1.2. 특성

전통적으로 전문가 시스템이 문제해결 방식으로 사용된 영역은 정적인 데이터가 주로 많이 사용되고 응답시간의 제한이 그리 중요하지 않은 영역이었다. 전문가 시스템 개발초기의 진단, 설계 및 시스템 구성 등의 영역에 적용된 시스템들은 이와 같은 특성을 보여주었다. 그러나, 인공지능 연구자들은 다음에서 설명하는 것과 같은 복잡한 문제들을 해결하기 위한 대안으로써 실시간 시스템과의 접목을 피하게 되었다[Gupta, 1985; Laffey, 1988; Mandelkern, 1992; Zhu, 1993].

1) 비단조성(Nonmonotonicity) : 센서를 통해 감지된 입력 데이터와 추론된 사실들은 프로그램의 전체 실행과정 동안 정적인 상태로 머무르지 않는다. 입력 데이터들은 시간이 지나면서 소멸하거나 타당성이 감소 또는 없어지게 된다. 왜냐하면 시스템과 관련된 사건들은 시스템의 상태를 변화시키기 때문이다.

2) 지속적 운용(Continuous Operation) : 대부분의 실시간 시스템들은 시스템 운용자나 어떤 결정적인 외부사건에 의하여 정지될 때까지 지속적으로 작동된다. 시스템의 일부또는 특정 부분에서 발생한 오류가 시스템의 기능을 정지시키지는 않는다. 실시간 시스템의 감시 또는 통제 장치는 쓰레기 모음(Garbage Collection)등의 문제에 깊은 관심을 기울여 지속적인 운용능력을 갖추어야 한다.

3) 비동기적 사건(Asynchronous Event) : 실시간 시스템에서 다루어 지는 많은 사건이나 행동들은 스케줄에 일치된다. 즉, 실시간 시스템은 체계적이고 예측가능한 형태로 데이터를 처리한다. 그러나, 실시간 시스템은 스케줄에 일치하지않고 발생하는 사건이나 비동기적 사건들도 효과적으로 다룰 수 있어야 한다. 뿐만 아니라, 발생사건들의 중요성이 각각 틀릴 수 있게 되는데 실시간 시스템은 사건들의 우선 순위에 따라 적절히 인터럽트하고 스케줄을 재조정할 수 있어야 한다.

4) 외부환경과의 인터페이스(Interface to External Environment)

일반적으로 실시간 전문가 시스템은 일련의 센서를 통하여 데이터를 수집한다. 그러나, 전통적인 전문가 시스템은 시스템 사용자 또는 운용자에게 데이터의 입력을 요구한다.

5) 불완전 데이터(Uncertain or Missing Data) : 데이터의 타당성은 시간의 흐름과 함께 감소 또는 소멸한다. 그러나, 때로는 센서의 성능 불량으로 타당성에 의문이 가는 데이터가 입력될 수 있다. 따라서, 실시간 전문가 시스템은 불완전하거나 타당성이 약한 데이터가 입력되었을때 이를 인식하고 적절히 처리할 수 있는 능력을 갖추어야 한다.

6) 높은 성능수준(High Performance) : 실시간 시스템의 활용에 있어 시스템의 성능수준은 아주 중요한 문제이다. 예를 들어 적 항공기의 출현을 감지하고 격퇴시키는 방공관제 시스템의 경우 적기 출현 감지와 대응행위 지시에 수백 milisecond의 범위내에서 정확히 식별하고 조치하여야 한다. 지금까지 급격히 변화하는 데이터를 다루어야 하는 복잡한 문제 영역의 경우 그 처리과정상의 병목 현상 제거 문제 때문에 많은 어려움이 있어왔다.

7) 시간적 추론(Temporal Reasoning) : 실시간 시스템 적용 영역에 있어 시간은 아주

중요한 변수이다. 특히, 실시간 시스템은 시간이 발생하는 순서는 물론 과거, 현재, 미래의 사건들에 관한 추론 능력도 갖추어야 한다.

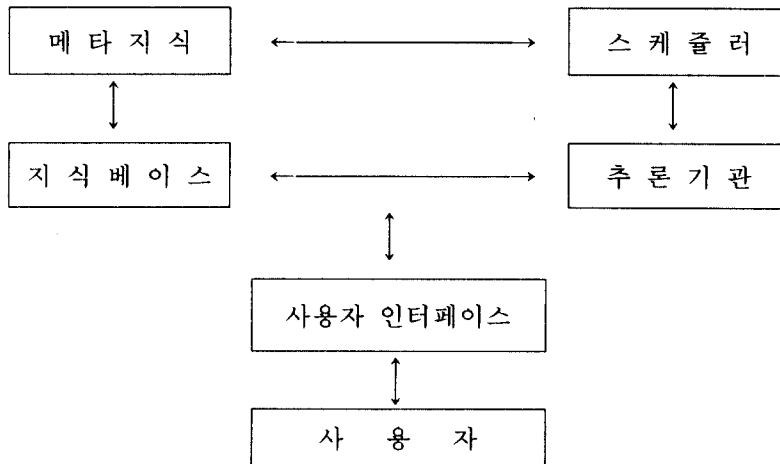
8) 관심의 집중(Focus of Attention) : 어떤 중요한 사건이 발생했을때, 실시간 시스템은 사용자 능한 자원을 그 사건의 해결에 우선적으로 집중할 수 있는 능력을 갖추는 것이 중요하다. 이러한 우선순위를 가진 문제 해결을 위하여 새로운 지식을 획득하고, 센서의 감지 대상이나 범위의 조정 및 데이터 분석 및 처리속도를 조정할 필요가 있을 것이다.

9) 응답시간의 보장(Guaranteed Response Times) : 실시간 시스템은 반드시 응답이 필요한 시간 범위내에 반응할 수 있어야 한다. 뿐만 아니라, 주어진 시간 제한내에서 가장 최선의 응답을 제공해야 한다.

10) 절차적 프로그램 요소와의 통합(Integration with Procedural Components): 실시간 전문가 시스템은 특히 기존의 실시간 모듈들은 데이터 압축, 신호처리, 특성추출, 특정 목적의 입·출력 통제 등의 처리를 할 수 있을 것이다.

2.1.3. 실시간 전문가 시스템의 구조

실시간 전문가 시스템의 구조는 그림 1에 나타난 바와 같이, 일반적인 전문가 시스템의 구조에 실시간 조건을 충족 시키기 위한 추론 스케줄러, 추론제어에 관한 정보를 제어하고 우선순위 개념을 통하여 추론시간을 단축시켜 줄 수 있는 메타지식이 추가된 형태를 갖춘다.



<그림 1> 실시간 전문가 시스템의 구조

실시간 전문가 시스템의 개발에 있어서 중요한 것은, 시스템의 외부환경과 실시간 통신 문제, 외부로부터 입력된 요구가 짧은 시간에 집중될 때 우선순위를 고려한 규칙의 우선 처리문제, 시스템의 오류 발생시 대응책, 병렬추론기법, 규칙충돌 해결책 등 실시간 조건을 충족시키기 위하여 처리해야 할 문제들은 다양하다. 전문가 시스템이 실시간 조건을 충족시킨다는 것은 질문을 입력한 후 정해진 시간 내에 질문에 대한 결과를 얻을 수 있어

야 함은 물론 정확한 결과를 사용자에게 제공할 수 있어야 한다.

이를 위하여 실시간 전문가 시스템에서는 특정 요구에 대한 추론이 규정시간 내에 끝나야 하고, 하나의 요구에 대한 추론 도중에 새로운 추론요구가 발생될 때, 우선순위의 부여를 통한 스케줄링이 필요하게 된다. 따라서, 실시간 전문가 시스템 설계시, 추론이 규정시간내에 종료될 수 있도록 적절한 지식베이스의 구성과 추론엔진의 추론기법을 사용하는 것이 필요하다.

한편, 실시간 전문가 시스템의 추론 스케줄러는 다음에 실행될 규칙을 결정하는 제어기능을 갖는데, 실행될 지식베이스의 규칙을 결정하기 위하여 규칙의 변수들의 조건을 평가하기 전에 새로운 추론요구들에 대한 반복적인 점검을 실행하게 된다. 실시간 전문가 시스템의 스케줄링 기법은 추론요구들에 할당된 우선순위를 부여하고 서비스할 추론요구들의 스케줄을 수립할 수 있도록 설계된다.

일반적으로 실시간 전문가 시스템의 성능에 영향을 주는 변수로는 태스크의 수, 실행시간, 소요자원수, 선점(preemption) 방식, 프로세서의 수 등이 있다. 실시간 전문가 시스템의 추론환경에서 나타나는 두드러진 특징을 살펴보면, 첫째, 외부 시스템으로부터의 추론요구는 독립적이며 비주기적으로 발생한다. 즉, 동적인 스케줄링(dynamic scheduling)을 요구하고, 둘째, 각 규칙들의 실행시간을 예측하는 것은 매우 복잡하고 어려운 문제로써 전문가 시스템의 설계시에 면밀한 고려가 요구되며, 셋째, 추론요구에 할당된 우선순위에 따라 적절히 인터럽트하고 스케줄을 재조정 할 수 있어야 한다.

실시간 전문가 시스템의 지식베이스는 문제 영역의 지식을 표현하는 지식베이스와 추론 제어 정보를 유지, 관리하는 메타지식으로 구성된다. 지식베이스와 메타지식은 선처리과정을 거쳐 완성되는데 다음과 같이 그 절차를 요약 할 수 있다. 먼저 일차적인 지식베이스가 구축되고나면, 지식의 우선순위 및 추이적 관계를 발견하고, 지식베이스의 정렬 및 추론비용을 산정하고 이를 토대로 지식베이스 및 메타지식을 완성하게 된다.

2.2. 실시간 전문가 시스템의 필요성

Turner[1986]는 “실시간 전문가 시스템을 사용하는 근본적인 이유는 사용자의 인지적 부담(Cognitive Load)을 덜어주거나, 인지적 부담의 증가없이 사용자들의 생산성을 향상시키는데 있다.” 고 하였다. 실시간 전문가 시스템의 바람직한 적용 영역은 기존의 방법으로 해결하지 못했거나 비현실적인 영역들로서 다음과 같은 문제 해결 분야가 관심을 끌고 있다.

즉, 인간의 인지적 부담 때문에 어려움에 처해 있거나, 효과적으로 모든 이용가능한 정보를 제어하는데 실패한 경우, 서로 상충되는 제약 요인들의 해결이 불가능한 경우, 고비용의 실수와 유망한 사업 기회의 포착 실패한 경우, 최적의 해를 찾기 위한 모든 관련 정보들을 동시에 효율적으로 조작할 수 없는 경우, 또는 신속한 문제 해결 요구에 응답을 주지 못 할 경우 등 다양하다.

앞에서도 언급한 바와 같이, 실시간 영역이 전문가 시스템이 제공하는 전문 지식을 획득하고 적용하는 능력으로부터 어떻게 혜택을 받을 수 있는가에 대해서는 많은 사례들이 뒷받침하고 있다. 예를 들면, 인공위성 운용과 통제에서 일정한 자격을 갖춘 사람을 구하

기가 점점 어려워지고 있는 상황에서 복잡한 상황을 해석, 평가하는 적절한 행동을 추천해 줄 수 있는 실시간 전문가 시스템은 많은 도움을 줄 것이다.

한편, 신속하고 적절한 의사결정이 중요한 증권 시장과 외환 시장에서 복잡하고 많은 양의 정보를 분석, 평가하여 신속히 최적의 의사결정을 내릴 수 있는 실시간 전문가 시스템은 아주 높은 가치를 가질 것이다.

III. 실시간 전문가 시스템의 개발과 활용

3.1. 실시간 전문가 시스템의 개발

실시간 전문가 시스템의 개발은 지식베이스와 추론기관의 설계에 있어 기존 전문가 시스템의 경우와 비교하여 훨씬 복잡하다. 실시간 전문가 시스템의 개발과 활용에 있어 고려해야 할 주요 이론적 연구문제들에 대하여 살펴본다.

3.1.1. 실시간 전문가 시스템 개발시 주요 고려사항

1) 성능수준(Performance)

규칙기반 시스템의 실행속도 문제가 지금까지 높은 성능수준과 실시간 응답을 요구하는 영역에서의 활용에 많은 장애가 되어왔다. Gupta[1985]의 연구에 의하면 현재 사용하는 규칙기반 시스템은 실행시간의 90%를 매칭 단계에 쓰고 단지 약 10%의 시간이 갈등 해결과 실행 단계에 쓰인다고 한다.

Rete 알고리즘은 다이내믹하게 변화하는 데이터를 수록한 지식 베이스 내의 규칙들의 매칭과정을 효율적으로 처리하도록 고안되었다. 이 알고리즘에서는 추론 과정에서 계속적으로 이루어지는 매칭, 선택, 실행의 단계에서 반복적 매칭을 줄이는 방안을 고려함으로써 효율성을 높이고자 한다. 대부분의 전문가 시스템은 추론과정이 진행되는 동안 작업 메모리의 내용을 거의 바꾸지 않는다. 따라서 추론과정 중에 불필요한 매칭이 많이 실행됨으로써 효율성을 떨어뜨리는데, 매칭 정보를 기억시켜 추론과정 내에서 작업 메모리 중 새로 변경된 것만을 비교 대상으로 고려하는 방법을 검토할 수 있다[Forgy, 1982].

Rete 알고리즘은 작업 메모리에 추가할 내용이나, 삭제할 내용, 변경할 내용을 충돌집합(Collict Set)에서 직접 변경하도록 한다. 그런 다음, 충돌집합 내에 포함된 규칙을 선택하여 실행하면 그것을 충돌집합으로 부터 삭제하고 그 나머지로써 다음 실행을 반복한다. 이렇게 함으로써, 작업 메모리에 대한 모든 규칙들의 반복적 매칭을 피함으로써 비교의 횟수를 크게 줄이는 효과를 가져올 수 있다.

Gupta, Forgy 그리고 Newell[Forgy, 1982; Gupta, 1985; Pulli, 1994]은 규칙 기반 시스템의 실행속도를 향상시킬 수 있는 다양한 방법들을 연구하였다. 그들은 규칙기반 시스템을 위한 컴퓨터의 설계시에 다양한 구조적 특성들을 검토하고 실행시간의 향상을 위한 병렬처리 개념을 제시하였다. 그들의 연구는 OPS5와 Soar에 의하여 개발된 몇개의 실제 규

칙기반시스템을 대상으로 수행되었다. 처음의 기대와는 달리, 병렬처리개념의 도입을 통한 실행속도의 향상이 10배 정도에 불과하다는 사실을 발견하였다. 이와같은 소폭의 상승효과에 그친 이유를 그들은 다음과 같이 설명하였다. 첫째, 지식베이스의 변경과 관련된 규칙의 수가 적고, 둘째, 규칙들의 처리요구와 관련한 변화정도가 크며, 셋째, 동기화 단계사이에 지식베이스내에 발생하는 수정, 변경의 횟수가 적다는 점이다.

더 나아가, 그들은 이와같은 제한적 실행속도의 향상효과에 대한 원인규명을 위해 병렬처리개념에 대한 새로운 처방과 대책을 모색하는 것이 필요하다고 강조하였다. 그들은 이와같은 시도의 일환으로 32 ~ 64개의 프로세서를 장착한 메모리 공유개념의 다중처리기(multiprocessor)를 적절한 시스템 구조로 제안하였다.

그러나, 지금까지 검토되어진 어떤 시스템들도 실시간 감시 또는 통제분야에 실질적으로 적용된 시스템이라고 볼 수는 없다. 사실 매실행주기마다 지식베이스에 대한 많은 변경을 수반하는 실시간응용에서는 Gupta[1985]에 의하여 제시된 10배의 향상효과 보다 훨씬 더 큰 실행속도의 향상을 병렬처리개념을 통하여 구현할 수 있다. 뿐만아니라, Gupta 등이 지적하였듯이, 이러한 연구결과는 다분히 당시 유행하는 프로그래밍 스타일에 의존적이고 프로그래밍 스타일이 변화하면서 그 타당성을 상실할 수도 있다. 더욱 연구결과의 해석을 어렵게 만드는 것은 시뮬레이션 결과들이 메모리공유 다중처리시스템에서 볼 수 있는 스케줄링, 동기화, 메모리경쟁등의 문제들을 포함하지 않는다는 점이다[Laffey, 1988].

2) 응답시간의 보장

현재 여러 특별한 기술들이 주어진 시간내에 원하는 응답을 제공할 수 있는 시스템의 개발에 사용되고 있다. 일반적으로, 실시간 시스템의 정상적 운용에 관한 책임은 프로그래머에게 있다. 즉, 프로그래머가 시스템을 개발하고, 운용하고, 실시간으로 작동하는지 여부를 검토하고, 지속적인 유지, 보수를 담당한다.

O'Reilly와 Cromarty[1985]에 의하여 논의된 바와 같이, 이와같은 시스템의 개발과 운용에 관련된 전반적 조율작업은 몇가지 중요한 문제점을 내포하고 있다. 첫째, 특별히 도입된 방법론들은 현재 프로젝트의 범위를 초과하여 수행되어 질 수 있다. 둘째, 문제의 사양과 데이터의 형태나 규모가 변하게 되면 시스템의 성능이 저하 될 수 있다. 셋째, 프로그래머의 조율과정은 아주 시간 소모적이다. 왜냐하면 프로그래머는 요구사항을 충족시킬 수 있는 해결점을 파악하기 위한 프로그램의 수정, 보완을 구체적인 제약없이 자유롭게 수행하기 때문이다. 마지막으로 시스템 평가를 위한 공식적인 기준이 결여되어 있고, 시스템이 우연적으로 실시간적 결과를 생성할 수 있다.

O'Reilly와 Cromarty[1985]의 연구에 의하면 정방향 추론을 사용하는 생성 시스템은 기하 급수적 시간을 요구하는 것으로 나타났다. 즉, 추론 트리의 길이가 증가함에 따라 'fire'되는 규칙들의 수가 기하 급수적으로 증가한다. 그들은 또한 역방향 추론에서도 추론 트리의 길이가 증가함에 따라 트리 노드의 수와 탐색 경로의 수가 기하 급수적으로 증가함을 발견하였다. 이와 같은 점을 고려할때 응답시간의 범위를 지정하는 것은 제한된 범위내에서 가능할 수 있을지라도, 최악의 경우 지정된 응답시간이 허용범위를 벗어날 가능성이 있다. 따라서, Rete 알고리즘 등과 같이 응답시간을 보장할 수 있는 알고리즘의 실행속도

개선 노력에 더욱 많은 관심과 노력이 필요하다[Natarajan, 1992].

한편, Frame 기반 언어들은 대표적인 지식 표현 방법의 하나로써 주로 Semantic nets 를 정의 하는데 사용되어진다. 'Instance' 관계는 일반적인 형태의 frame으로 부터 상속된 속성들과 디폴트(default)값들을 설정하고 실행시간에 주어진 형태의 모든 'Instance'들을 검색하고 처리하는데 사용된다. 그리고, 'Is-a' 관계를 추가하고, 이를 활용하여 전체 프로그램 계층 구조에서 frame의 유형을 설정함으로써, 객체지향적 프로그래밍을 지원할 수 있다.

상속 네트워크는 단순 상속의 경우 다음과 같이 정의할 수 있으며 단일 루트(root)를 가진 트리 구조를 형성한다. 즉, 각 클래스(class)는 하나의 상위 클래스를 가지며 단지 두 개의 relation으로 'Is-a'와 Instance만이 허용된다. 이와 같은 두개의 relation에 제한되는 점을 개선하게 위하여 새로운 언어들은 사용자들이 frame간의 복수상속관계(multiple inheritance)를 허용하여 하나이상의 상위 클래스를 갖도록 하여준다. 이러한 복수상속관계는 더욱 융통성있는 지식 표현을 가능하게 해 주지만, 새로운 문제점을 내포하고 있다. 즉, 상속 네트워크가 예측하기 어려운 방향성 그래프가 되고, slot으로 부터 값을 탐색할 때 분석과정의 지수적 성질 때문에 실시간-응용프로그램에 적합하지 않을 수 있다.

3) 의사결정의 신뢰성 제고

많은 실시간 감시·통제 응용프로그램에서는 다양한 업무들을 처리하기 위하여 주어진 시간내에 실행되어야 하므로 충분히 사려 깊은 의사결정을 고려할 수 없다. 그러므로, 가능한 시간내에 최선의 의사결정을 도출하는 것이 중요하다. 이러한 실시간 프로그램에의 의사결정의 효과성 제고를 위한 연구들을 살펴보면, Horvits(1988)에 의하여 제시된 Decision Analytic Techniques, Winston(1984)의 Progressive Deepening, Wright(1986) 등의 Progressive Reasoning, Sorrells(1985)의 Time-Constrained Inference Network, 그리고 Michalski와 Winston[Laffey, 1988]의 Variable Precision Logic등을 들 수 있다.

3.1.2. 실시간 전문가 시스템 개발도구

1) 실시간 전문가 시스템 개발도구 현황

본 논문에서 소개된 많은 개발 도구외에도 현재 수백개의 개발 도구들이 상업적으로 소개되어 활용되고 있다. 이들 중에서 Picon과 G2 등이 실시간 감시 및 통제등의 시스템 개발에 적극적으로 활용되고 있는 실정이다. 가장 광범위하게 사용되고 있는 Tool을 살펴보면 표 1과 같다.

실시간 영역은 시간적 요인에 대한 의존성 때문에 복잡하고 다이내믹한 문제들을 제기한다. 실시간 전문가 시스템은 기존의 전통적인 전문가 시스템 활용에서 제공하지 못했던 시스템 성능 요구를 충족시켜야 한다. 그러나, 현재 사용가능한 기존의 개발 도구들은 일반적으로 다음과 같은 이유들 때문에 실시간 전문가 시스템 응용에 적합하지 않은것으로 판단된다[Allard, 1987; Laffey, 1988; Mandelkern, 1992].

첫째, 미 국방성의 DARPA(Defence Advanced Research Project Agency)연구에 의하

면 개발도구들의 실행속도가 너무 느리고,
 둘째, 개발 도구들이 시간적 추론(Temporal Reasoning)기능을 제공하지 못하거나 제한적이며,
 셋째, 개발 도구들과 전통적인 소프트웨어와의 효율적인 통합이 어려우며,
 넷째, 개발 도구들이 주요사건에 대한 집중능력이 미비하거나 결여되어 있으며,
 다섯째, 개발 도구들이 비동기적 사건의 처리 능력을 소유하지 않고 있고,
 여섯째, 개발 도구들이 소프트웨어/하드웨어 인터럽트를 처리하는 방법이 제공되지 않으며,

<표 1> 실시간 전문가 시스템 개발도구

개발도구 명	주요 특성	비고
Activation Framework	The system supports priority utilization and focus of attention	The Real-Time Intelligent Systems Corp.
BBN/Patterns	It provides real-time recognition of patterns and events in time-series data	BBN Systems & Technology
Blackboard Objects(BLOBS)	Written in PROLOG and POP11 Blackboard concept with object oriented system	
The Experimental Expert Systems Flight Status Monitor (EESFSM)	Developed by NASA Research Center Applied to flight control system	
Expert Controller	The shell can run 5000 rules/second An automotive hood-stamping process control application (General Motors)	UME Corp.
Forth	Built for McDonnell Douglas The system can run 30,000 rules/second on the Novix chip	Forth-based Prolog system available
Fuzzy Inference Chip	Developed by AT&T Bell Lab. The system can run 80,000 fuzzy logic inferences/second and suitable for real time use in intelligent robot systems and decision making areas of command and control	

G2	Built in Common Lisp The system supports temporal reasoning, focus of attention, truth maintenance, real time scheduling, and interprocess communication	Gensym Corp.
The Hybrid Expert System Controller (HEXSCON)	Written in Pascal The system combines conventional logic programming with expert system technology for use as a real time process controller	
ObjectGEODE	An object-oriented tool for s/w development teams working on real-time, distributed application for client-server networks It operates on a range of workstations, including Sun SPARC, HP7000, IBM RS/6000 and Digital ALPHA	available from VERILOG
ONSPEC Superintendent	written in Pascal runs under FlexOS real-time operating system	
Personal Consult Plus (PC Plus)	Developed by Texas Instruments PC Plus supports reasoning about trends reflected in a history of previous sensor data	run with PC On-line
The Process Diagnostic System (PDS)	written in FranzLIPS on a VAX running VMS. PDS provides online, real-time diagnosis of machine processes	
The Process Intelligent Control (PICON)	implemented in LISP on the TI Explorer and LMI Lambda/Plus machines The first commercial real-time expert systems tool for developing process control applications	also extended to work on VAX / VMS.
The Procedural Reasoning System (PRS)	implemented on a Symbolics 3600 Lisp Machine PRS was built to perform reasoning and planning in dynamic & uncertain worlds	

The Real-Time Expert System Club Users(RESCU)	RESCU is a real-time expert system for production plants The System is applied to Quality Control Activity with participation of club users	
RTworks	The System facilitates the acquisition, anlysis, distribution, and display of real-time data It can run 20,000 rules/second on a Sun SPARC station	
Violet and Annie	built in C Violet provides the intelligent interpretation and Control of Complex instruments for Vibration-based mechanical health monitoring Annie supports process monitoring and mechanical health diagnosis	Intelligent Application, Inc.
XNET	XNET is a network management expert system shell The System monotor and analyzes network data online	
The Yorktown Expert System Lang. one (YES/LI)	Implemented in OPS5 and MacLisp. It is general-purpose language for developing real-time, rule-based programming applications	run on the VM and MVS/XA operating systems

일곱번째, 개발도구들이 인간이외의 외부자극을 통한 입력을 효율적으로 처리할수 없고, 마지막으로, 개발 도구들이 제한된 응답시간을 충족할 수 있는 능력이 미흡하고, 실시간 처리에 맞도록 설계된 하드웨어 상에서 운용되지 못하고 있다. 기존의 개발 도구들을 실시간 영역에 적용하는 것은 Prolog를 숫자처리 프로그램에 적용하거나 Fortan을 기호처리 응용에 사용하는 것과 같다. 실시간 응용에 적합하지 않는 개발 도구들을 사용하는 것은 아주 의미없는 일이다. 따라서, 실시간 영역에 적용될 수 있는 완벽한 능력을 갖춘 전문가 시스템 개발 도구들의 개발에 대한 연구가 필요하다.

2) 실시간 전문가 시스템 개발도구의 선택

실시간 전문가시스템의 성능수준을 사용자가 원하는 수준으로 확보하기 위해서는 적절한 개발도구의 선택이 중요하다. 실시간 영역에 적용되는 전문가시스템이 일반적인 전문가시스템 개발도구에 의하여 개발될 때, 이미 앞에서 언급하였듯이 제한적인 성능수준에 머무르거나 원래 기대했던 요구사항을 충족 할 수 없게 된다. 실시간 전문가 시스템 개발도구의 사용이 필요한 응용분야를 결정하는데 도움이 될 수 있는 주요 평가항목을 살펴보면 다음과 같다[Mandelkern, 1992].

- 1)규칙 처리 속도로 측정되는 성능수준 요구사항이 일반 전문가시스템 개발도구로 충족 될 수 없을 때,
- 2)추론과정이 단순히 현재 데이터의 값에 의존하기 보다는 일련의 시간적 흐름에 따라 이루어지는 경우,
- 3)적용되는 응용분야가 규칙, 사건, 또는 데이터들을 특정 시간대에 연결하여 실시간 clock과 통합시켜야 할 경우,
- 4)시스템이 비동기적으로 발생하는 사건이나 데이터를 처리 할 수 있는 능력이 요구 될때,
- 5)주어진 응답시간내에 신뢰성있는 시스템의 결과가 중요 할 때,
- 6)데이터 입력이 사용자를 통해서 보다는 센서등의 외부 프로그램 source로 부터 이루어 질 때,
- 7)우선순위의 배정과 중요한 사건에 대한 자원의 집중이 요구 될 때,
- 8)시스템 유지및 보수, 메모리 손상, 또는 쓰레기 모음등에 대한 걱정없이 24시간 지속 적으로 운용되고, 기존의 실시간 모듈과의 효율적인 통합이 필요 할 때 등이다.

3.2. 실시간 전문가 시스템 활용현황

현재 실질적으로 산업현장에서 활용되고 있는 대부분의 전문가 시스템은 약 만개 이상의 규칙을 포함하고 있으므로, 순차적인 추론방법을 채택한 전문가 시스템의 구축으로는 탐색, 추론, 설명능력 제공시에 신속성이 결여되기 때문에 병렬처리방법을 적용한 실시간 전문가 시스템의 필요성이 증대되고 있으며 활발하게 이에대한 연구가 진행중에 있다. 이러한 실시간 전문가 시스템을 구축하기 위해서는 병렬운영체제, 병렬 컴파일러, 병렬 인공지능어 및 병렬 스케줄링 기법의 개발 등이 병행하여 연구되어야 한다[Chen, 1985; Kiskis, 1990; Leinweber, 1986; O'Reilly, 1985].

병렬처리방법은 순차적 처리방법에 비하여 시스템의 신속성이 매우 증대되기 때문에 각종 인공지능 시스템에 적용할 필요성이 있으며, 전문가 시스템의 수행능력을 제고하기 위하여 이의 적용이 크게 강조되는 분야이다. 왜냐하면, 모든 인공지능 시스템의 응용분야에는 탐색(search)기법이 핵심적인 기법으로서 중요한 역할을 수행하므로, 가능하면 신속하게 원하는 정보를 얻어야 하기 때문이다.

또한, 전문가 시스템은 대화식(interactive) 시스템이므로 의사결정권자에게 신속하게 원하는 정보를 제공하여야 하며, 인공지능 시스템의 대표적인 특징인 추론을 신속하게 할 수 있도록 병렬처리 개념을 전문가 시스템에 적용하는 것은 매우 중요한 의미가 있다. 이러한 전문가 시스템의 병렬화는 특히 군사적 응용시스템에는 필수적인 고려요소이다. 왜

냐하면, 대부분의 군사적응용 전문가 시스템은 실시간 시스템이기 때문이다. 즉, 지역적으로 분산되어 있거나 국부적인 데이터 및 사건을 시스템으로 받아들여서 주어진 시간 내에 처리해서 응답하는 시스템으로서, 군사적으로는 표적 할당, 무기 할당, 지휘관의 의사결정 지원을 위한 전문가 시스템 그리고 사회적으로서 전형적인 데이터 수집과 트랜잭션 (Transaction) 처리분야, 공정제어 및 공장자동화 분야 등에 활용되고 있다.

본 논문에서는 기존의 연구활동의 결과로써 AI Magazine, IEEE Expert, AI Expert 등에 발표된 활용사례 및 제품들을 중심으로 자료를 수집 이를 특징적인 몇개의 분야로 구분하였다. 표 1에서 보는바와 같이 항공 공학, 의학, 공정 제어, 로봇 공학등 병렬처리개념의 도입이 절실하고 높은 수준의 신뢰도와 신속한 응답시간이 요구되는 분야를 중심으로 발전해 왔는데 이와같은 추세는 타 영역에도 점차 확산 적용될 것으로 전망된다.

<표 2> 실시간 전문가 시스템 활용현황

항공분야	통신분야	의학분야	공정제어	로봇공학	품질관리 및 스케줄링
AIRPLAN	ECESIS	BABY	ALFA	Autonomous	RESCU(shaw)
EPES	ERIK(f)	CAPS	CEALMON	Vehicle Guidance(f)	REDS(Hadavi)
ESSOC			REALM		
EXIMU	HANNIBAL	EEG Analysis	COOKER(f)	HERMIES(f)	
ExpertNavigator-or			DiagnosticE.S.(f)		
FLES	News Wire Monitoring	FORTES	ESCORT	Ping Pong Player(f)	
HEAT			Polysomnographer		
LES		VM	LMA		
L · STAR			MCM		
Malfunction Recovery			REACTOR		
NAVEX			Rotary Cement		
Pilot's Associate			Kiln Supervixor(f)		
PREMON			STOCHASM		
SCARES			YES/MVS(f)		
SECURE					

IV. 결 론

본 논문에서 밝혀진 조사결과에 의하면, 지금까지 전통적인 전문가 시스템에 적용해왔던 접근방법 보다 훨씬 복잡하고 어려운 실시간 전문가 시스템의 개발과 활용에 많은 노력을 기울여 왔다. 실시간 문제해결에 있어서도 특히, 중요한 정보를 소홀히 하거나 너무 늦게 대응 한다든지, 정보의 양이 너무 과다할 때 느끼는 공포감 등의 인간의 인지적 경향에 대한 극복과 해결이 주요 관심사가 된다.

지금까지 구현된 실시간 전문가 시스템의 분석 결과를 토대로 전문가 시스템 관점에서 중요한 문제들을 검토하고, 실시간 전문가 시스템이 갖춰야 할 바람직한 특성들과 실시간 전문가 시스템 연구의 주요 이론적 문제점 요약하면 다음과 같다.

1) 수치계산과 기호 처리계산의 효율적 통합: 지금까지 데이터 압축, 신호처리, 특성추출 등과 같은 다양한 실시간 문제 해결을 위하여 많은 알고리즘들이 개발되었다. 이러한 알고리즘들이 전문가 시스템의 기호처리 모듈과 효율적으로 통합되는 것이 필요하다.

2) 지속적 운용: 실시간 전문가 시스템은 오류가 발생하더라도 중단없이 지속적으로 운용되어야 한다. 실시간 시스템에서는 오류가 발생되었을때 기능이 정지되면 엄청난 결과가 초래된다. 따라서, 실시간 시스템이 오랜기간 동안 지속적으로 운용되기 위하여 쓰레기 모음(garbage collection)등에 세심한 관심을 기울여야 한다.

3) 관심 집중 메카니즘: 실시간 시스템은 특정 규칙이 적용될때 그 정황 또는 배경을 분석하고판단하는 능력을 가져야 한다. 또한, 어떤 중요한 사건이 발생하였을때 시스템의 자원을 집중 투입할 수 있는 기능이 필요하다.

4) 인터럽트 처리능력: 현재 수행중인 처리에 대한 인터럽트 기능과 더욱 우선적인 사건이 처리되고 난 후 계속적으로 처리가 진행될 수 있도록 조절할 수 있는 비동기적 메시지의 관리능력이 필요하다.

5) 예측가능 능력: 전문가 시스템의 운용 상태를 충분히 예측 가능해야 한다. 즉, 주어진 시간제한 내에서 전문가 시스템의 응답시간이 측정 가능해야 한다.

6) 시간적 추론 능력: 실시간 시스템의 지식 표현에 있어 시간적 관계의 표현을 허용할 수 있는 기능이 중요하다. 즉, 시간적 데이터에 대한 검색, 관리 및 통계적으로 검증할 수 있는 기능이 필요하다.

7) 타당성 평가 능력: 일반적으로 경험적 데이터는 시간이 지남에 따라 질적인 변화를 갖게 되는 함수관계가 존재한다. 이와 같은 사실은, 센서를 통하여 데이터 입력을 받는 환경에서 특히 많이 발생한다. 데이터의 타당성이 감소함에 따라, 현재 시점에서는 타당성이 없는 데이터에 기초한 모든 추론을 제거할 수 있는 타당성 검토 메카니즘이 필요하다.

실시간 영역은 전문가 시스템의 개발과 응용에 새롭고 흥미로운 가능성을 제공한다. 특히, 극심한 경쟁환경의 변화와 고도로 발달한 정보화 사회속에서 정보처리의 신속성과 고도의 신뢰성을 갖춘 실시간 전문가 시스템의 도입은 어느때보다도 중요하다고 하겠다. 그러나, 실시간 전문가 시스템의 확산과 관련하여 아직도 해결해야 할 많은 문제점들이 연구과제로 남아 있다.

References

- Ali, M., and Scharnhorst, D. A. "Sensor-Based Fault Diagnosis in a Flight Expert System," Proceedings of the Second Conference on Artificial Intelligence Applications, Washington D. C.: IEEE Computer Society, 1985, pp. 49-52.
- Allard, J. R., and Kaemmerer, W. F. "The Goal/Subgoal Knowledge Representation for Real-Time Process Monitoring," Proceedings of the Sixth National Conference on Artificial Intelligence, 1987, pp. 394-398.
- Brooks, M. IBM ES to Aid in Mainframe Management. Applied Artificial Intelligence Report, 4(6): 17, 1987.
- Chen, D. C. "Progress in Knowledge-Based Flight Monitoring," Proceedings of the Second Conference on Artificial Intelligence Applications, 1985, pp. 441-116.
- Ennis, R. L., Klein, D., Milliken, K., Schor, M., Greismer, J., Hong, S., Karnaugh, M., Kastner, J., and Van Woerkom, H. "A Continuous Real-Time Expert System for Computer Operations," IBM Journal of Research and Development, 30(1), 1986, pp. 14-28.
- Expert Systems Resource Guide, AI Expert, April 1995, pp. 26-36.
- Faulk, S., Bracket, J., Ward, P., and Kirby, J., Jr., "The Core Method for Real-Time Requirements," IEEE Software, September 1992, pp. 22-33.
- Forgy, C. L. "RETE: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem," Artificial Intelligence, 19, 1982, pp. 17-37.
- Gopinath, P., Bihari, T., and Gupta, R., "Compiler Support for Object Oriented Real-Time Software," IEEE Software, September 1992, pp. 45-50.
- Gupta, A., "Parallelism in Production Systems: The Sources and the Expected Speed Up," Expert Systems and Their Applications, 5th International Workshop, 1985, pp. 26-57.
- Hadavi, K., Hsu, W., Chen, T., and Lee, C., "An Architecture for Real-Time Distributed Scheduling," AI Magazine, Fall 1992, pp. 46-56.
- Hamilton, M. "SCARES-A Spacecraft Control Anomaly Resolution Expert System," In Proceedings of the 1986 Expert Systems in Government Conference, 1986, pp. 436-443.
- Kiskis, D., and Shin, K., "A Synthetic Workload for Real Time Systems," Proceedings of Workshop on Real Time Operating Systems and Software, 1990, pp. 77-81.
- Laffey, T., Cox, P., Schmidt, J., Kao, S., and Read, J., "Real Time Knowledge Based Systems," AI Magazine, Spring 1988, pp. 27-45.
- Leinweber, D., and Gidwani, K. "Real Time Expert System Development Techniques and Applications," Proceedings of WESTEX-86: IEEE Western Conference on Knowledge-Based Engineering, 1986, pp. 69-77.

- Mandelkern, D., "Ingredients for Real-Time Expert Systems," *AI Expert*, July 1992, pp. 31-35.
- Natarajan, S., and Zhao, W., "Issues in Building Dynamic Real-Time Systems," *IEEE Software*, September 1992, pp. 16-21.
- New Product Announcements. 1987. *AI Magazine* 8(2): 130-131.
- O'Reilly, C. A., and Cromarty, A. S. " "Fast" is not "Real-Time" in Designing Effective Real-Time AI Systems," *Applications of Artificial Intelligence*, 1985, pp. 249-257.
- Park, J. "Toward the Development of a Real-Time Expert System," *Journal of Forth Applications and Research*, 4(2), 1986, pp. 133-154.
- Pisano, A. D., and Jones, H. L. "An Expert Systems Approach to Adaptive Tactical Navigation," *Proceedings of the First Conference on Artificial Intelligence Applications*, Washington D. C.: IEEE Computer Society, 1984.
- Product News, *AI Magazine*, Summer 1996, pp. 94-115.
- Pulli, P., and Heikkinen, M., "Concurrent Engineering for Real-Time Systems," *IEEE Software*, November 1994, pp. 39-44.
- Shin, K., Kandlur, D., Kiskis, D., Dodd, P., Resenberg, H., and Indiresan, A., "A Distributed Real-Time Operating System," *IEEE Software*, September 1992, pp. 58-68.
- Stoyenko, A., and Halang, W., "Extending Pearl for Industrial Real-Time Applications," *IEEE Software*, July 1993, pp. 65-74.
- Turner, M. "Real Time Experts," *Systems International*, 14(1), 1986, pp. 55-57.
- Weisbin, C. R. "Real-Time Control: A Significant Test of AI Technologies," *IEEE Expert*, 2(4), 1987, pp. 16-17.
- Winston, P. H. *Artificial Intelligence*, 2nd ed., Addison-Wesley, 1984, pp. 129-131.
- Wright, M., Green, M., Fiegl, G., and Cross, P., "An Expert System for Real Time Control," *IEEE Software*, March 1986, pp. 16-24.
- Zhu, J., Lewis, T., Jackson, W., and Wilson, R., "Scheduling in Hard Real Time Applications," *IEEE Software*, July 1993, pp. 54-63.