

전문가 시스템 (Expert System)

최 창 근*

1. 서론

다가오는 21세기는 인공지능 산업시대가 될 것이라고 일컬어지고 있다. 심지어 일부 공상가들은 인공지능에 의해 인간 지능의 완전 대체를 성급하게 논의하고 있기도 하다. 그러나 이런 일은 아직 요원한 일일 뿐이고 다만 인공지능 기술분야에서 지금까지 가장 성공적인 경우로 전문가 시스템(Expert System)을 들 수 있는 정도이다. 전문가 시스템은 극히 제한된 전문분야에서 전문인과 비슷한 수준의 일을 수행할 수 있는 프로그램을 일컫는다. 이는 필연적으로 전문 지식의 프로그램화를 전제로 하므로 지식베이스 전문가 시스템(KBES: Knowledge-Based Expert System)으로 일컬어지기도 한다. 전문가시스템의 개발에 관해서는 많은 문헌, LISP, OPS5, KEE 및 ART 등의 전문가 시스템의 시스템 개발용 도구(Tool), 그리고 심볼처리 전용기기인 리스트 머신등의 하드웨어들이 보급되어 있지만, 실용적인 전문가 시스템의 개발을 위해서는 아직 더 많은 시간과 노력이 요구되고 있다.

전문가(인간)가 어떤 작업(예:구조계산)을 수행하기 위해서는 교육과 기본적인 훈련을 통하여 얻게 되는 깊은 지식(Deep Knowledge) 뿐만 아니라, 과거에 비슷한 문제를 해결했던 경험등에 의해 얻어진 경험지식(Surface Knowledge)이 필요하다. 이러한 전문가의 지식을 프로그램화 하였다가 새로운 문제에 직면했을때 전문가 시스템을 통해 이러한 프로그램화된 지식을 활

용함으로써 경험이 적은 비전문가도 전문가에 준하는 작업능률을 얻을 수 있게된다. 구조공학 분야에서도 성공적으로 개발된 전문가 시스템은 계획(Planning), 해석(Analysis) 및 설계(Design) 등에 효과적으로 이용될 수 있다. 그러나 전문가 시스템의 개발에 대한 이론적인 면에 대하여 현재의 실용적인 측면에서의 개발현황은 세계적으로 아직 미흡한 실정이므로 우리나라에서도 선진국과의 차이가 더 벌어지기 전에 이 분야의 기술개발에 보다 많은 참여가 요망되는 시점이라 할 수 있다.

2. 문제영역의 선택

전문가 시스템이라고 하여 인간생활의 모든 분야에서 컴퓨터가 인간 전문가를 대신하리라는 기대는 아직 공상적이라 할 수 있다. 전문가 시스템의 개발은 현재로서는 전문성이 확연하게 정의될 수 있는 좁은 문제영역에서 시스템 구성을 위한 도구(Tool)가 문제풀이에 아주 적합한 경우에만 성공할 수 있고, 실제의 개발은 오랜동안의 반복에 의하여 이루어진다. 따라서 전문가 시스템이 모든 문제에 대한 해결책은 될 수 없고, 제한된 상황(범위)에서만 유용하다고 할 수 있다. 이러한 전문가 시스템의 구성요소들의 세부적인 역할과 그 상호관계를 나타내면 보통 다음 그림과 같이 표현된다. 즉 전문분야의 지식을 담고 있는 지식베이스, 현재의 문제풀이 상황을 기록하는 Context, 전문분야의 지식을 제어하는 방법을 나타내는 추론기관으로 구성되고, 추가되

*정희원, 회장, 한국과학기술원 교수, 공학.

는 기능으로 전문가 시스템의 확장을 쉽게 하기 위한 지식 획득 시설, 사용자에게 편의를 제공하기 위한 사용자 인터페이스 및 설명기능등이 있다.

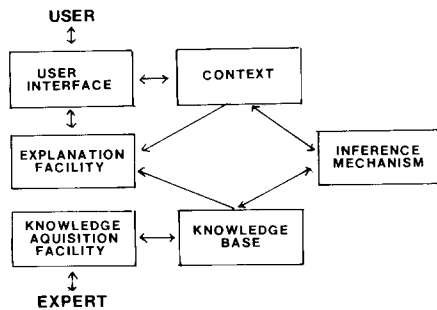


그림 전문가 시스템의 구성

○ 지식베이스(Knowledge Base)

지식베이스에는 해결하려는 분야에 관한 지식들(전문성이 있는 지식(Deep Knowledge)과 경험적인 지식(Surface Knowledge)을 포함하고 있다. Deep Knowledge는 역학의 정적평형조건 등과 같은 기본적인 원리의 지식이며, Surface Knowledge는 기본적인 원리로부터 경험에 의하여 생겨난 것이다. 구조해석 및 부재의 설계는 Deep Knowledge에 가까우며 여러가지 구조물의 형상에서 후보를 선정하는 문제들은 Surface Knowledge로 구분될 수 있다. 보통 Deep Knowledge는 기존의 알고리즘식 프로그램 언어인 FORTRAN, BASIC, C, PASCAL 등을 이용하여 함수(Function) 및 프로그램으로 표현되며, Surface Knowledge는 생성시스템(Production System)에서 사실(Facts)과 규칙(Rules)으로 나타내어진다. 사실은 선언적인 지식으로 나타나는 데 이는 물체, 개념등과 그들의 관계를 표현하는데 이용된다. 규칙은 조건부분(Condition 또는 Premise Part)과 실행부분(Action 또는 Consequence Part)으로 나누어 지는데 실행부분이 작용하기 위해서는 조건부분이 참(True)값이 되어야 한다.

○ Context

Context는 문제해결의 현재상태를 반영하는 사실(Facts)들로 구성되어 있다. 이러한 Context의 조직은 문제영역의 성질에 따라 달라지게 되며, 해당문제의 해결과정중에 수시로 변하거나 추가 및 삭제가 된다. 이 내용들은 추론기관(Inference

Mechanism)에 의하여 문제해결의 다음 단계를 결정하는데 사용된다.

○ 추론기관(Inference Mechanism)

추론기관은 지식을 다루는 방법을 나타내는 사항인데, 이는 지식베이스에 내장된 지식을 이용하여 현재의 Context 상태를 조정한다. 전형적인 지식베이스 전문가 시스템은 다음의 과정을 반복하게 된다.

- (1) 현재의 Context와 지식베이스의 패턴을 비교하여 적용가능한 모든 규칙을 찾는다.
- (2) 적절한 하나의 규칙을 선택한다.
- (3) 그 규칙의 실행부분을 적용하여 Context의 내용을 변경시킨다.

○ 지식획득모듈(Knowledge Acquisition Module)

지식획득모듈은 전문가(인간)와 KBES와의 Interface로서의 역할을 한다. 이는 전문가의 지식을 KBES에 입력하거나 필요한 경우에는 내장된 지식들을 변경하는 작업등을 할 수 있게 한다. 즉 사용자가 아주 편한 에디터를 사용하여 파일에 저장된 규칙을 수정하거나 추가하고, 또는 추론과정에서 지식베이스를 조정하기도 한다.

○ 사용자 인터페이스(User Interface)

사용자 인터페이스 모듈은 사용자에게 KBES의 사용을 편리하게 한다. 이러한 모듈은 KBES의 실행을 위한 기본적인 명령문을 제공하고, 사용자가 입력한 내용을 이해하고, 이러한 정보제어에 대한 결과를 그 문제에 맞게 설계된 문장(Syntax)이나 컴퓨터 그래픽을 이용하여 사용자에게 대화식으로 전달한다.

○ 설명기능(Explanation Facility)

설명모듈은 KBES의 일에 대한 추론과정을 사용자에게 제시해 주는 기능을 한다. 이는 KBES가 실행과정에서 현재의 상태까지 이르게된 경위라던지, 현재의 질문이 왜 필요한지에 대한 설명을 해주게 된다. 가장 보편적인 설명모듈의 형식은 결론이 나오기 까지의 유추과정을 거슬러 올라가는 형식(Retrospective Reasoning)을 취하는 데, 이는 사용자로부터의 질문에 대한 추론 결과를 어떻게 혹은 왜 선정하였는지에 대한 설명과, 추론에 관계된 일련의 법칙들(Rules)을 제시하여 준다. 이러한 설명모듈을 추가함으로써

추론된 결과에 대한 사용자의 의구심을 해소시킬 수 있고 전문가 시스템이 전문가의 모습으로 역할을 할 수 있게 된다.

3. 전문가 시스템의 구축

3.1. 지식의 표현도구

전문가 시스템을 구성하는 기본적인 요소는 지식의 표현과, 지식의 처리방법(보통 추리과정이라함)이다. 전문가 시스템에 내장되는 지식의 표현을 위하여 사용되는 프로그램 언어는 다음과 같이 분류될 수 있다.

절차식 언어와 논리 언어:인공지능형 언어를 구성하기 위한 기본적인 바탕이 되는 언어로서 전자로는 C, LISP등이 있고, 후자로는 PROLOG 등이 있다.

생성시스템용 언어:앞에서 언급한 언어와 다른점은 프로그램의 수행이 순차적이지 않고 무순위로 작동된다는 것이며 데이터에 따라 적용이 좌우된다. 여기서는 지식의 단위로 규칙(Rule 또는 Production)이 사용되며, 매칭(Matching)과 실행을 위해서는 Recognize-Act 사이클의 제어관격을 가진다. 이들 예로는 EMYCIN, OPS5, CLIPS등을 들 수 있다.

객체 지향적 언어:객체(Object)의 부류에 데이터와 처리 과정을 저장하여 지식을 처리하는 방법으로 예를 들면 FLAVORS, SMALLTALK 등이 있다.

복합형(Hybrid)언어:가장 높은 단계의 지식표현 기법으로서 규칙과 객체의 표현이 모두 가능한데, 이들 예로서는 ART, KEE, Knowledge Craft 등이 있다.

3.2. 지식의 처리

지식의 처리란 3.1 절에서 언급한 지식의 표현 형태를 이용하여 적절한 해를 구하기 위하여 지식베이스에서 사고하는 추론과정을 일컫는다. 이러한 처리과정은 크게 규칙(Rule)과 프레임(Frame)의 두 형태로 분류할 수 있다.

규칙(Rule 또는 Production):엄격한 객체 지향적 언어를 제외한 거의 모든 전문가 시스템은

규칙이라는 데이터로부터 결론을 이끌어 나가는 전진추론(Forward Chaining 또는 Data-driven)과 이와 반대로 가능한 해를 가정하고, 그 가정을 지원해주는 데이터를 검증해내는 후진추론(Backward Chaining 또는 Goal-driven)의 형태가 있다. 그러나 실생활에서 일어나는 대부분의 문제는 위 두 형태가 적절히 조합된 광범위한 추론형태를 요구하는 문제들이므로, 수많은 대안들을 검토하여 주어진 상황에 대한 최적해를 찾기 위하여 가능한 해는 전부 구해보는 것이 필요하다. 이러한 방법으로는 관점추론(View-Point 또는 Hypothetical Chaining)이 있다.

프레임(Frame 또는 Object):전문가 시스템의 프로그램 기법으로서 크게 두가지 형태로 분류된다. 첫째는 객체의 속성을 묘사하는 Slot들의 집합으로 표현되는 프레임이 있고, 둘째는 객체간의 관계를 나타내는 의미망(Semantic Network)이다.

이상에서 언급한 내용을 종합하면 가장 바람직한 전문가 시스템은 전진 및 후진 그리고 관점추론의 규칙이 객체에 저장된 정보를 자유롭게 교환할 수 있는 것이라 할 수 있다.

3.3. 불확실한 지식의 처리

정보는 항상 완전한 것이 아니므로 전문성을 요구하는 많은 문제들은 불확실한 지식도 처리하는 것이 필요하다. 이러한 문제들을 처리하기 위해 구축된 전문가 시스템들은 여러가지 방법으로 이를 구현하고 있으며, 현재 보급되는 일부 도구(Tool)들은 이러한 능력을 가지고 있다.

4. 효율적인 지식의 관리

여러분야에서의 전문가 시스템의 성공사례는 더 광범위하고 복잡한 분야에서도 실현가능성을 시사한다고 할 수 있다. 그러나 실용가능한 시스템의 구축과정에서 직접적으로 접하게 되는 주요한 문제는 지식베이스에 있는 규칙과, 작동 메모리(Working Memory)에 있는 데이터 요소가 증가함에 따른 관리의 어려움이다. 또한 그로 인한 추론시간의 연장등으로 실시간(Realtime) 영

역에서 문제해결의 능력이 저하될 우려가 있으므로 전문가 시스템의 효율적인 구성을 통하여 메모리 관리능력을 높여야 한다.

5. 시스템 개발상의 문제

전문가 시스템에 관한 이론적인 문헌들은 많이 보급되어 있으나, 실제 개발의 제반 과정에 대한 언급은 별로 없다. 본문에서는 실제로 전문가 시스템의 개발시 접하게 되는 여러 사항에 관한 문제들을 기술하고자 한다.

1) 기존의 소프트웨어의 개발때와는 달리 전문가 시스템의 개발은 그 요구조건이 유연하기 때문에 개발방법이 미리 정의되기 어렵다. 따라서 원형(Prototype)의 설정, 얻어진 결과와 기대치와의 비교, 평가과정등을 반복하면서 점진적으로 개발해 나가야 한다.

2) 규칙 기반(Rule Based) 시스템에서는 기존의 프로그램으로도 문제를 효과적으로 처리할 수 있는 경우가 있다. 이 때는 기존의 프로그램과 전문가 시스템 간에 정보교환 즉, 기존의 프로그램에 의해 만들어진 메모리 내의 데이터를 공유하거나, 규칙의 RHS(행위부)에서 기존의 프로그램의 함수를 호출하는것 등이 가능해야 한다.

3) 사용자와의 인터페이스 문제는 대화식으로 작동되는 전문가 시스템에서는 매우 중요하다. 이러한 부류에 속하는 기능으로는 윈도우 시스

템, 마우스 조정기능 및 메뉴양식등이 있다.

4) 전문성의 영역과 그 정도, 사용자 수준을 선정하여야 한다.

5) 만약 선택된 문제가 이미 구현된 시스템과 비슷하면 기존의 것을 최대한 이용하고, 되도록이면 가장 수준높은 언어를 이용한다.

6. 맺는말

본 기사에서는 전문가 시스템의 구축에 관한 제반사항, 즉 전문가 시스템에 대한 개략적인 소개, 문제영역의 선택, 전문가 시스템의 올바른 모델, 지식의 관리요령 및 시스템 개발상의 문제들에 대하여 기술하였다. 이러한 전문가 시스템은 특정분야 전문가의 지식을 컴퓨터에 내장함으로써 전문가가 회귀한 분야에서 그 지식을 쉽게 획득하거나, 그 분야의 초보자를 교육시키거나, 전문가의 부재시 일의 능률을 유지시키기 위한 목적등의 실효성이 점차 인정되어 가면서 그 개발노력도 증대되고 있다. 그리고 전문가 시스템의 개발은 기존 프로그램 방식과는 달리 개념의 구체화 작업의 연속으로, 반복적인 과정을 통하여 이루어지므로 다소 시행착오적인 일이라 할 수 있다. 따라서 시스템의 구축에는 많은 시간과 노력이 필요하다. 구축된 전문가 시스템은 그 전문성을 높이기 위하여 지식베이스내에 새로운 지식의 Update가 부단히 계속되어야 한다.