기하 문제 학습을 위한 동적 추론 체계
(A Dynamic Inferential Framework for Learning Geometry Problem Solving)

국 형 준 †
(Hyung Joon Kook)

요 약 수리나 과학 영역의 학습은 원리 이해와 응용을 위주로 함에도 불구하고 기존의 교육용 소프트웨어 제품들은 단순 주입식이나 단답식의 학습을 지원하는 것이 대부분이어서 높은 학습 성과를 기대하기는 어려운 실정이다. 인공 지능 연구에서 지식 표현 체계나 탐색, 추론 기법이 학습기 설계에 도입되어 증명기, 모의 실험기 유형의 학습기 연구에는 상당한 진전을 보아 왔으나 여전히 실용적 수준이라 할 수는 없고 특히 문제 해결을 지원하는 학습기는 설계 모형조차 제시되지 못하고 있다. 본 연구가 설계한 기하 문제 학습기는 학습과 병행하는 동적 추론을 구사한다. 실시간 문제 해결을 지원하기 위한 정보 구성 요소로서 명제, 가설 및 연산자에 의해 문제 공간을 정의하고 이들의 생성과 검증을 추론의 주요 대상으로 하는 대화식 문제 학습의 메카니즘을 탐구하였다. 성취한 결과로서 기하 문제 해결에서 필수 불가결한 요소임에도 불구, 기존 시스템이 간과해 왔던 대수 처리를 위한 일련의 추론 전략을 연계적으로 구사함으로서 실용성있는 문제 학습기의 설계 모형을 얻었다. 제안 모형은 물리, 전자 회로 등 타 과학 영역의 문제 학습기 설계에도 적용될 수 있다.

Abstract In spite that the main contents of mathematical and scientific learning are understanding principles and their applications, most of existing educational softwares are based on rote learning, thus resulting in limited educational effects. In the artificial intelligence research, some progress has been made in developing automatic tutors based on proving and simulation, by adapting the techniques of knowledge representation, search and inference to the design of tutors. However, these tutors still fall short of being practical and the turor, even a prototype model, for learning problem solving is yet to come out. The geometry problem-solving tutor proposed by this research involves dynamic inference performed in parallel with learning. As an ontology for composing the problem space within a real-time setting, we have employed the notions of propositions, hypotheses and operators. Then we investigated the mechanism of interactive learning of problem solving in which the main target of inference involves the generation and the test of these components. Major accomplishment from this research is a practical model of a problem tutor embedded with a series of inference techniques for algebraic manipulation, which is indispensable in geometry problem solving but overlooked by previous research. The proposed model is expected to be applicable to the design of problem tutors in other scientific areas such as physics and electric circuitry.

1. 서 론 및 연구 배경

정보화 사회로의 급속한 진입과 함께 교육용 소프트웨어 개발에 대한 관심과 수요도 높아지고 있다. 하지만 현재 대부분의 교육용 소프트웨어의 주종이 영어를 비롯한 외국어 학습과 유아용의 비교적 단순한 제품인 점에서 볼 수 있듯이, 교육에 관해서는 아직도 컴퓨터 기술이 극히 보조적이고도 제한적인 수준에서만 활용되고 있는 실정이다. 그 원인으로는 기존의 교육용 소프트웨어의 종류가 그리 다양하지 못하며 제공하는 학습 성능 또한 높지 않다는 점을 꼽을 수 있다. 특히 수리, 과학

· 이 논문은 1996년도 한국과학재단의 핵심과제(과제 번호: 961-0901-004-2) 연구비에 의하여 연구되었음.
† 종신회원 : 세종대학교 컴퓨터공학과 교수
kook@sejong.ac.kr
논문접수 : 1999년 3월 24일
심사완료 : 1999년 9월 30일

영역에서는 더욱 그러한데 이 분야에서의 사용 가능한 제품은 일부 유아용의 단순 주입식이나 단답식 학습기를 제외하곤 전무한 형편이다.

높은 학습 수요가 존재함에도 불구하고 수리, 과학 분야의 고차적 학습을 적극 지원할 소프트웨어가 부족한 것은 해당 학습 영역의 복잡성을 적절히 수용할 소프트웨어 설계 기술이 아직 정립되지 않았다는 데 기인한다. 즉 수리, 과학 학습은 암기나 단순 주입보다는 원리 이해와 응용을 위주로 한 훈련에 기반하는 특성을 가진다. 때문에 단순히 공식 원리만을 주입식으로 강의하는 방식으로는 학습 성과를 기대하기 어려우며, 이 보다는 원리를 응용한 다양한 문제의 해결에 관해 학습하는 것이 더욱 중요한 것이다. 그간 수리, 과학 영역의 학습기 모형은 주로 인공 지능 분야에서 활발히 연구되어 왔다. 인공 지능 연구에서 얻은 지식 표현 체계나 탐색, 추론 기법이 학습기 설계에 도입되어 증명기, 모의 실험기 유형의 학습기 연구에는 상당한 진전을 보아 왔으나 아직까지 실용적 수준의 문제 학습기 모형은 제시되지 못하고 있다.

학습 성과와 실용성이 높은 문제 학습기 개발 필요성에 부응하여, 본 연구는 특히 기하 학습에 활용 가능한 문제 학습기 모형 개발을 목표하였다. 연구의 주요 목표로서 첫째 다양한 문제 해결 학습을 지원하며, 둘째 학습자와 상호 작용성이 제공되며, 셋째 기하 영역이외의 수리, 과학 영역에도 적용 가능한 문제 학습기의 모형 개발을 추구하였다. 본 논문은 이에 관한 연구 결과를 다음 순서로 제시한다. 이어 지는 2 장에서는 먼저 관련 연구를 소개한다. 다음 3 장에서 본 연구가 제안하는 문제 학습기의 설계 요소를 소개한 후 이의 구현 방법론을 4, 5, 6 장으로 나누어 설명한다. 7 장에서는 본 문제 학습기가 처리 가능한 학습 문제들의 범위와 예를 제시하고, 마지막으로 8 장에서 결론과 향후 연구를 제시한다.

2. 관련 연구

기하, 물리, 전기 회로는 초, 중, 고등 학생이 반드시 습득해야 하는 교과 영역이지만 대부분의 학생이 실지 학습 과정에서 상당한 어려움을 겪는 분야이다. 따라서 이러한 분야에 적용 가능한 컴퓨터 교수 체계를 설계 및 구현하는 것은 학문적인 연구 가치 외에도 실제적인 효용성이 높다고 할 수 있다. 이 장에서는 기하, 물리, 전기 회로 분야 등 과학 영역의 기존 연구에서 제안된 지식 표현 방법 및 문제 해결에 관한 인지 모델과 교수 방법론을 살펴본다.

과학 영역에서의 지식 표현에 대한 연구는 많은 인지 과학자 및 인공지능 연구자에 의해 발전되었다. Greeno[10]는 기하 영역에서 동일한 지식을 다른 문제 영역에 응용하기 위한 지식의 표현 형태에 대해서 분석하고 이것이 영역간의 문맥적 유사성(analogy)을 분석함으로 획득될 수 있다고 설명하였다. 본 연구에서는 과학 영역 공통의 지식을 표현하고 사용할 수 있게 하는 문제 해결 체계를 제시함으로써 Greeno가 제기한 영역간에 이전될 수 있는 의미 있는 지식을 명시적으로 표현할 수 있게 한다. Reif[19]는 과학영역에서 지식 표현을 일반성과 명확성을 정확히 표현하는 공식적인 것과 특수한 사례에 대해서 빠른 추론을 가능하게 하는 비공식적인 것으로 유형화하였다. 이에 반해 Larkin[13]은 FERMI라는 지식 표현 체계를 통해 상이한 인지적 차원에서 상이한 지식을 합성하고자 하였다. FERMI는 스키마로 표현된 지식들을 의미적인 계층으로 분리하고 영역 고유의 지식과 영역 독립적 지식사이에 차별화된 사실들(예: 과학 원리)과 전략 지식을 인코드한다. 스키마에 의한 지식 표현은 이외에도 Anderson[1]의 ACT 인지 모델에서도 중요한 역할을 한다.

Forbus[8]와 White[22]가 제시한 과학 영역의 지식 표현은 질적 모델을 추구한다는 점에서 앞서의 연구들과는 구별된다. Forbus는 자신이 개발한 QPT (Qualitative Process Theory)에서 열류나 운동같은 물리 시스템의 동적인 개념을 질적으로 모델링하기 위한 이론을 소개했다. 이 이론에서는 FERMI에서와는 개념을 달리하는 질적 스키마로서 물리 객체의 속성을 표현하고 그들에 대한 추론을 제공한다. 특히 QPT는 이러한 표현 구조를 통해 수량 공간에 대한 질적인 표현과 아울러, 시간 의존적인 방식에 의한 동적 시스템의 인과적 묘사를 가능하게 한다. White는 질적 모델을 통해서 동적 절차에 대한 인간의 사고 과정을 모방하였다. 질적 모델은 전기 회로내의 변화에 따른 회로의 상태 변화를 관찰하고 이해하기 위한 수단으로서 많은 장점을 가진다. 그러나 질적 모델링에만 의존한 학습 체계는 인과적 형식의 순박한 지식만을 대상으로 하는 경향이 있기 때문에, 질적 추론 학습을 위한 하나의 방식으로 사용되는 것은 바람직하나, 아직까지 선언적, 절차적 지식 기반의 문제 해결기를 바탕으로 한 학습을 배제할 만한 우월성은 불분명하다고 할 수 있다[2].

앞서 언급한 지식 표현과 연계되어 과학 영역의 추론 모델 역시 다양한 기법이 시도되어 왔다. Gelernter[4]의 Geometry Theorem-Proving Machine은 공리들에 기반한 역방향 추론으로 기하 증명 문제를 해결하는 시

스템이다. 이 시스템은 단순한 역방향 추론에 문법적 대칭성(syntactic symmetry) 인식이라는 방법을 추가하여 탐색 공간을 축소시킨다. 그러나 문제 해결 과정의 논거나 검증에 대한 논의를 등한시하고 주로 기계 효율적인 문제에만 집중된 문제 해결 방법을 제공하고 있다. 따라서 기하 증명기로서의 선구적 의미는 있으나 현재 본 연구가 제안하는 대화식 문제 학습기의 설계에는 적용 곤란하다.

Anderson[1]은 ACT 이론을 고등학교 기하 영역에 적용하여, 학생이 증명 문제를 어떤 계획아래 해결하고 학습하는지를 설명하였다. ACT 이론은 인간의 인지 과정을 표현하는 하나의 인지 모델로서 생성 시스템인 절차적 지식과 스키마 형태의 선언적 지식으로 구성된다. 절차적 지식은 선언적 지식의 번역(knowledge compilation)에 의하여 생성되는데, 실제 문제 상황에 따라 다양한 각도에서 해석될 수 있는 잇점이 있다. ACT 모델이 제시한 지식 번역 방법은 본 연구에서도 추구된 연산자 자동 생성과 상통하는 면이 있다[23].

McDougal[16]은 기하 문제 해결 시스템에서 사용되는 전문지식의 특성을 반영한 POLYA라는 시스템을 설명했다. 이 특성은 [12]에서 관찰된 "기하학에서 전문가는 목표를 고려하지 않고 주어진 정보 즉, 전제 조건으로부터 추론을 하는 경향이 있다"라는 사실에 기반하는데, McDougal은 이러한 특성을 반영하기 위해서 Koedinger[11]와는 달리 문제 공간과 다이아그램을 인식하는 과정을 병행하는 정방향 추론에 문제 해결을 의존한다. 따라서 POLYA의 문제 해결의 전과정은 다이아그램 인식이라는 커다란 과제를 포함한다. 이와 같은 기법은 다이아그램을 수반하지 않는 학습 내용의 ITS로의 이전성이 부족하다. 본 연구와 비교하면 트리거를 사용한 해결 전략에 있어서 유사성이 있지만 다이아그램에 의존된 정방향 추론인 점에서 큰 차이가 있다.

SOPHIE[5, 7, 21]는 전기 회로 모의실험실 환경에서 학생에게 문제 해결에 사용되는 기법들을 교수하는데, 문제 해결을 위해서 SPICE라는 모의실험기를 사용하여 영역 지식을 인코딩한다. 이 모의실험기는 수학 계산을 수행하는 것으로 인간이 문제를 해결할 때 주로 심볼에 기초하는데 반해 수치적인 행동 양식을 갖는다. 따라서 전기 회로 영역의 인간 지식 모델은 포함하지 않고 수학적 모델에 기초한 추론을 수행한다. Anderson [2]은 이런 문제 해결 방법을 'black box' 모델이라고 부르는데, black box 모델에 기초한 문제 해결에서의 내부적인 추론 행위는 인과 관계에 기반한 추론에 바탕을 두지 않으므로 설명 기능이 부족하다. 따라서 교육 영역에서 사용하기에는 매우 부적절하다. SOPHIE의 확장된 버전 SOPHIE-II에서는 이런 약점이 보완된 형태의 인과 모델을 정립하였으나, 근본적으로 학습 내용 면에 있어서 비교적 초보적 수준의 질적 추론에 기반한 모의실험을 제공한다. 이에 반해 본 연구가 제시하는 문제 학습기 모형은 영역 모델로서의 기하 공식 원리 해석에 기초한 추론과 교수를 위주로 설계되어 있어, 영역 지식의 학습이 직접적으로 제공된다 할 수 있다.

3. 문제 학습기 설계 요소

본 연구는 기하 문제 학습의 각 단계에서 대화식 학습을 지원하는 학습기 체계의 설계를 목표하였다. 그러나, 이와 같은 목표의 성취를 위해 미리 선정된 모든 문제에 대해 그 해결 과정과 정답을 미리 데이타베이스 형태로 구축하고 이를 기반으로 교수를 시도하고자 접근한다면 여러 가지 문제점이 따르게 된다. 첫째, 문제마다 그 해결 과정까지 저장해야 하므로 많은 양의 기억 장소를 사용함에 따라 데이타베이스의 확장을 저해한다는 점이다. 둘째, 이렇게 해서 저장된 시스템 해결 과정과는 종종 상이한 경로를 취하는 학습자 해결 과정에 대해 유연한 대처가 불가능하다는 점이다. 실제 많은 기하 학습용 문제들은 하나 이상의 해결 경로를 가지며, 이 가운데 어느 경로가 '최적해'라고 할 수 없는 경우가 많다. 또한 학습자의 수준에 따라 해결 경로가 다를 수 있으며, 학습기는 이에 관해 적응할 필요도 있는 것이다. 이에 대처하기 위해 문제에 관한 가능한 모든 해결 경로를 저장하고 이에 기반하여 학습자의 해결 과정을 추적하는 방식은, 각 문제에 대한 저장 데이타 양이 더욱 늘어난다는 점과 비대한 탐색 공간을 대상으로 한 추론 부담 때문에 학습의 실시간 성능이 저하되는 점 등 비효율을 수반한다. 특히 이 방식으로는 미리 저장된 문제에 관해서만 교수할 수 있으므로, 저장된 문제 밖의 학습자가 질의하는 문제에 관해서는 교수가 불가능하다는 점이다. 기존 대부분의 컴퓨터 학습기는 시스템에 저장된 문제에 관해서만 교수가 가능한 체제로 설계되어 학습의 범위는 시스템의 데이타베이스에 의해 제약된다. 그러나 이와 같은 제약을 벗어나 다양한 문제가 시도될 수 있을 때 진정한 의미에서의 문제 학습기라 할 수 있다.

앞서 말한, 문제의 해결 과정을 문제와 함께 미리 저장하는 방식의 단점들을 피하기 위해, 새로운 방식의 접근이 필요하다. 즉, 문제의 해결을 학습기 실행의 실시간에 행하는 방식이다. 구체적으로, 본 연구의 학습 체계는 미리 구축된 공식 원리 지식 기반을 사용하여, 학

습자가 선택 또는 질의한 문제에 대한 해결을 학습의 실시간에 행하는 방식으로 설계되었다. 학습자는 시스템의 데이타베이스로부터 문제를 선택할 수도 있고, 또는 학습자 자신의 문제를 입력하여 그 해결 과정을 학습할 수 있다. 그 결과, 상이한 학습자 수준에의 적응, 학습자 해결 과정에 동적으로 적응된 힌트 또는 설명 제공, 데이타베이스 밖의 학습자 질의 문제 허용, 기억 장소 사용 절감에 따른 데이타베이스 확장성 증대 등 많은 긍정적 효과를 얻는다.

다시 말해 본 연구가 제안하는 설계 원리의 핵심은 내부적으로 구축된(builtin) 지식 체계가 외부에서 공급된 지식을 적절히 해석, 처리함으로서 교수 행위를 발생시킨다는 점이다. 내부 구축된 지식 체계는 기하 영역과는 독립된 문제 해결 제어 구조와 학습 지원 전략을 보유한다. 반면, 외부에서 공급되어야 할 지식은 기하 영역의 공식 원리를 말한다. 기하 문제 학습기는 위의 두 가지 지식을 활용한 동적 추론에 기반하여 교수하게 된다. 즉, 본 연구가 말하는 동적 추론은 앞서 말한 바와 같이 학습자의 상이한 수준에 실시간에 적극적으로 적응하기 위해 도입한 일련의 추론 체계를 지칭하는 개념이다.

동적 추론에 기반한 학습의 또 다른 이점은 대화식 문제 학습 환경의 구축을 가능케 한다는 점이다. 기존 대부분의 문제 학습기는, 문제가 제시되고 학습자가 답을 하면 이에 대해 시스템이 맞다 틀리다로 피드백 해주는 식의 단답식 학습모드를 취한다. 그러나 원리 이해라든가 단계적 풀이 방법의 습득을 주목적으로 하는 기하 영역에 있어서, 단답식 학습은 높은 학습 성과를 가져다 주기 어렵다. 동적 추론에 의한 문제 학습은 이와 같은 제약을 극복할 수 있는 적절한 대안이 된다. 구체적으로 다음과 같다. 학습자는 주어진 문제 해결에 관한 자신의 가설을 단계별로 입력한다. 학습자 가설 입력은 학습자가 문제 해결을 위한 한 단계의 명제를 입력하는 것을 말한다. 간과할 수 없는 점은 학습자의 문제 해결이 각자의 수준에 따라 상이한 경로를 진행하며 학습자에 따라 다양한 수준의 논리적 비약을 수반하기도 한다는 점이다[3, 12, 15]. 이때 동적 추론에 기반한 학습기는 학습자의 가설이 성립하는 것인지를 검증하여, 옳고 그름에 관해 긍정 또는 부정으로서 응답할 수 있게 된다. 즉, 해결 경로가 시스템에 미리 저장되어 있는 것이 아니라 동적 추론에 의하여 생성되기 때문에 학습자의 정당한 추론을 모두 실시간에 검증, 허용할 수 있다. 따라서 높은 수준 학습자가 구사하는 논리 비약도 정당하다면 허용하므로, 시스템의 미리 정해진 수준의 해결 방식만을 강요하는 우를 범하지 않게 된다. 결과적으로 학습자의 해결 과정에 대해 긍정 및 부정만을 표시하며 지켜 보는 방식을 취함으로써 시행 착오를 통한 문제 학습이 이루어 지도록 한다.

이어지는 장들에서는 이 장에 제시된 문제 학습기 설계 요소가 본 연구가 설계한 기하 문제 학습기(CyberTutor)의 구축에 어떤 방식으로 구현되었는지를 정보 구성 요소, 동적 추론 체계, 그리고 학습자 가설 검증으로 나누어 설명한다.

4. 정보 구성 요소

본 연구에서 설계한 문제 학습기의 정보 구성 체계는 매 문제마다 초기화되는 작업 메모리의 내용으로 설명될 수 있다. 작업 메모리는 특정 문제와 관련된 명제(propositions), 가설(hypotheses), 그리고 문제 해결을 위한 연산자(operators)로 구성된다. 다음은 이들에 각각에 대한 설명이다.

학습기는 외부로부터 학습 영역의 공식들과 학습 문제를 공급받는다. 교사에 의해 공급되는 공식들과 문제들은 학습기의 공식 기반(Model Base)과 문제 집단을 각각 구축하는데 사용된다. 공식은 "평행하는 두 선분사이의 엇각의 크기는 같음", "삼각형의 내각의 합은 180도임" 등과 같은 명제로 표현되며, 문제 역시 그 내용이 명제로써 표현된다. 학습기에 의한 문제 학습은 데이타베이스에서 선택되거나, 혹은 학습자에 의해 주어진 문제를 대상으로 진행된다.

학습 문제가 선택되면, 해당 문제 해결을 위한 연산자(operators)들이 문제 로드 초기에 연산자 생성기에 의해 자동으로 생성된다. 일반적으로, 과학 학습 문제는 상황론적인 내용이 주어지고 여기에 더하여 몇 가지 제약 조건이 주어지는 것이 보통인데, 본 학습기는 전자를 구조적 명제(structural propositions)로, 후자를 주어진 명제(given propositions)로 구분하여 취급한다. 연산자 생성기는 이 가운데 구조적 명제들을 공식 기반에 의해 해석함으로써 현 문제 해결에 관련된 연산자, 즉 공식 적용 규칙(rewriting rules)들을 생성한다. 자동 생성된 연산자들과 주어진 명제들은 추론기의 작업 메모리의 초기 내용을 구성한다. [23]은 이 과정에 관한 상세한 내용을 기하 학습용 저자 인터페이스와 함께 설명한다.

4.1 명제와 가설

명제는 문제 해결의 현 시점까지 알려진 단위 사실들을 말한다. 예를 들어, 기하에서 "AB=5 (선분 AB의 길이)", "AB ∥ CD (두 선분의 평행)" 등과 같은 것이다. 가설은 아직 검증을 필요로 하는 사실들을 말한다. 가설

은 "∠ABC=45?", "AB+CD=180?" 등과 같이 어떤 사실 검증을 위한 것과 "X=?" 등과 같이 어떤 미지수의 값을 구하기 위한 것의 두 종류가 있다. 이들은 문제 해결의 여러 시점에서 학습기에 의해 검증의 목표로써 주어진다. 명제에는 문제 해결을 위해 초기에 주어진 조건들과, 문제 해결이 진행되면서 추가되는 것들이 있다. 명제의 성립 근거 공식이나 명제간의 의존 관계는 힌트 제시, 설명 등 다양한 교수 행위에 중요한 정보로서 활용된다. 이 같은 정보를 제공할 목적으로 각 명제는 몇 가지 속성을 가지는데, 해당 명제의 근거 공식명, 해당 명제를 지지하는 명제들의 리스트, 그리고 해당 명제가 지지하는 명제 등이 그것이다. 따라서 이들 속성 정보에 의해 작업 메모리의 명제들은 의존 그래프(dependency graph) 형태를 취하게 된다. 또한 명제 집단은 문제 해결에 따라 단조적으로(monotonically) 증가하는 것이 아니라, 학습기의 제어에 따라서는 작업 메모리에서 삭제되기도 한다. 추론기는 이에 따른 속성값을 동적으로 갱신함으로써 의존 그래프의 논거를 문제 학습의 어느 시점에서나 유지한다.

4.2 연산자

연산자는 문제 학습이 시작될 때마다 명제들와 함께 로드되며, 현 문제 해결과 관련된 공식 적용 규칙들(rewriting rules)로 구성된다. 연산자들은 기하 공식을 직접 반영하는 규칙들로서 추론기에 의해 선택적으로 적용된다. 구체적으로, 연산자에는 '<조건>⇒<결론>' 형식과 조건부없이 '<결론>' 형식의 두 가지 유형이 있다. 조건부는 명제들의 논리곱(conjunction)으로, 결론부는 단일 명제로 구성된다. 예를 들어, "AB=DE∧∠ABC=∠DEF∧BC=EF ⇒ △ABC≡△DEF (SAS 합동 법칙)"와 같은 조건부 형식의 연산자는 조건부의 세가지 조건이 모두 작업 메모리에 존재하여야만 적용될 수 있는 연산자이다. 반면, "∠ABC+∠BAC+∠ACB=180 (삼각형 내각의 합)"과 같은 무조건 형식의 연산자는 작업 메모리의 내용에 관계없이 적용될 수 있는 연산자이다.

연산자들은 장기 메모리의 규칙들에 의해 정 또는 역방향으로 적용된다. 정방향으로 적용될 경우 결론부의 명제를 작업 메모리에 추가하게 되고, 결론부의 명제를 가설(hypothsis)로 하여 역방향으로 적용될 경우 조건부의 명제들을 부목표(subgoal)로 설정하게 된다. 연산자의 적용 시점 및 방향에 관한 결정은 추론기에 의해 완전히 제어된다.

문제의 직접적 해결과 관련된 지식을 연산자 집단에 의해 표현함에 따른 가장 중요한 잇점은, 연산자에 의해 문제 해결의 여러 단계를 세분화하고 각 단계의 실행을 정밀히 제어할 수 있게 됨으로서 대화식 문제 학습 환경이 가능해진다는 점이다. 이에 관한 자세한 설명은 다음 장에서 제시된다.

5. 동적 추론 체계

작업 메모리의 초기화에 이어 진행되는 문제 학습은 추론기의 제어에 의해 이루어 진다. 학습기는 CLIPS[24]에 의해 작성된 추론 관련 부문이 Visual C++에 의해 작성된 인터페이스기와 연계하여 Win9X상에서 동작하도록 구현되었다. 작업 메모리, 규칙 메모리 및 추론 엔진으로 구성되는 추론기의 작업 메카니즘에 대해 간략히 소개하면 다음과 같다.

추론기는 '연속 구동'되는 방식에 의해 작동한다. 즉, 추론기는 한 문제에 대한 학습의 시작과 함께 작업 메모리가 초기화되면서 구동(start)되어 현 문제에 대한 학습이 계속되는 동안 재구동(resume)과 정지(pause)를 되풀이하면서 작업한다. 구체적으로, 추론기는 초기 구동시에는 먼저 정방향 추론을 수행하여 작업 메모리를 확장한다. 이후 학습자와 대화가 시작되면서 부터는 학습기로부터 검증하고자 하는 가설이 전달될 때마다 재구동되어, 초기 구동된 이후 계속 갱신, 유지된 작업 메모리에 기반한 추론을 행하며, 추론 결과를 반환한 후 정지 상태에 들어가는 '재구동→추론→정지' 싸이클을 반복한다.

문제 해결은 명제에 관한 추론을 위주로 수행된다. 명제에 관한 추론은 자료 유도적(data-driven) 정방향 추론이나 목표 유도적(goal-driven) 역방향 추론으로 수행할 수 있다. 여기서 목표란 주어진 가설, 즉 학습기가 추론기 재구동과 함께 전달한 가설, 그리고 차후 추론 과정에서 발생한 부목표(subgoal)로서의 가설들을 말한다. 그러나 명제에 관한 자료 유도적 또는 목표 유도적 추론 어느 한 쪽 혹은 이 둘만으로는 기하 문제 학습을 효과적으로 수행할 수 없다. 기하 문제 해결은 이외에도 대수적 추론이라든가 미지수의 처리 등이 적절히 구사될 때 가능한 경우가 대부분이기 때문이다. 따라서 가설에 관한 추론을 보완하기 위해, 대수에 관한 추론 및 미지수에 관한 추론을 구사한다. 이들 세 가지의 추론 전략은 모두 규칙 메모리의 일부로서 구축된다. 이들은 추론 엔진에 의해 해석되어 추론의 각 시점에서 상이한 우선 순위에 의해 동작한다. 이어지는 절에서는 각 추론 전략의 내용을 상세히 설명한다.

5.1 명제 추론

명제에 관한 추론은 일상적인 정방향 및 역방향 추론 양식을 취한다. 추론기가 초기 구동될 당시, 즉 문제가

로드되고 학습자와의 상호 작용이 시작되지 않은 시점에서는 정방향 추론을 행하여 작업 메모리의 내용을 확장한다. 구체적으로, 연산자 $P_{1,2,\cdots,N} \Rightarrow P$ (즉, $P_1 \wedge P_2 \wedge \cdots \wedge P_N \Rightarrow P$)가 존재하고 $P_1, P_2, \cdots, P_N$과 동치인 명제가 각각 존재한다면, 명제 P를 생성하여 작업 메모리에 추가한다.

가능한 정방향 추론이 모두 행해지고 학습자와의 상호 작용이 시작되면, 추론기는 '재구동→추론→정지' 싸이클에 진입한다. 이때부터의 추론은 학습기에 의해 전달받은 가설을 목표로 한 역방향 추론을 위주로 수행한다. 즉, 목표가 가설 H일때, 작업 메모리에 가설 H와 동치인 명제 P가 존재하면 가설 H가 만족된 것으로 하고 정지한다. 그러한 명제가 존재하지 않으나 연산자 $P_{1,2,\cdots,N} \Rightarrow P$ 이 존재하고 가설 H와 명제 P가 동치인 경우, $P_1, P_2, \cdots, P_N$를 각각 부목표로 설정하여 진행하고 모든 부목표가 만족된 경우에 명제 P를 작업 메모리에 추가한다. 그러한 연산자가 존재하지 않거나 부목표 가운데 일부가 만족되지 않는다면 이 추론 전략은 가설 H를 만족시키는데 실패하게 된다.

정방향 또는 역방향 추론의 성공으로 새로운 명제 P가 작업 메모리에 추가될 때, 명제들의 속성에 관해 몇 가지 갱신 작업이 이루어 진다. 첫째, 명제들의 의존 관계에 대한 기록 유지이다. 즉, 명제 P를 지지하는 명제들의 리스트와 명제 P가 지지하는 명제를 P의 속성에 각각 기록함으로써 명제들간의 의존 그래프를 갱신한다. 둘째, 현재 적용된 연산자의 근거 공식명을 명제 P의 해당 속성에 저장한다. 이와 같은 속성 정보들은 다음 장에 설명할 학습자 가설 검증에 활용된다.

명제에 관한 학습기의 추론 전략과 일상적 추론과의 차이점은 매칭시에 나타난다. 즉, 두 명제간의 '동일성'이 아닌 '동치성'만으로 충분히 매칭이 일어나야 한다는 점이다. 이는 과학 영역의 특성에서 기인한다. 예를 들어, 수식 명제 "X+Y=10", "X=10-Y", "X+Y-10=0", 또는 평행 관계 명제 "AB∥CD", "CD∥AB" 등은 각각 동일하지는 않으나 동치 관계에 있다. 이 같은 동치성에 대한 고려는 필연적으로 기하 문제 학습기의 추론의 복잡성을 증가시키기 때문에 기존 시스템에서는 학습자의 자유로운 입력을 제한하는 형식의 상호 작용 환경만을 제공함으로서 이를 배제하여 왔다. 본 연구에서는 이와 같은 특성을 충분히 고려하여 추론 부문을 구현함으로서 기존 시스템에 비해 자유도 높은 상호 작용 환경을 실현한다.

5.2 대수 추론

대수(algebra)에 관한 추론은 현 목표가 작업 메모리의 명제들(즉, 주어진 명제 및 유도된 명제들)로부터 대수적으로 유도 가능한 것인지 검사하는 것이다. 간단한 예를 들자면, 가설 "X=30?" 또는 "X=?"는 작업 메모리에 명제 "Y=30" 및 "X=Y"가 존재한다면, 대수적으로 추론 가능한 것이 된다. 사실 또는 미지수 두 가지 유형의 가설 모두 방정식 해결기에 의존하여 처리하지만 그 처리 방식에는 약간 차이가 있다. 먼저, 목표가 "X=?"과 같은 미지수인 경우, 그 값이 작업 메모리의 명제들에 의해 대수적으로 해결되면 목표 미지수 X와 해결 값(예를 들어 '30')의 등식으로 구성된 명제 "X=30"를 생성, 작업 메모리에 추가한다. 다음, 목표가 "X=30?"과 같은 가설인 경우, 작업 메모리의 명제들로부터 목표 가설이 대수적으로 유도될 수 있는지 검증하여 성공하면 목표 가설을 작업 메모리에 새로운 명제 "X=30"로서 추가한다. 만약, 목표 가설이 항진임이 판명된 경우에는 'tautology'를, 목표 가설이 작업 메모리의 명제들과 모순임이 판명된 경우에는 'contradiction'을 각각 반환하고 추론을 정지한다.

대수 추론 과정은 자연히 방정식 해결을 위한 시간을 필요로 한다. 대수 추론을 병행함에 따른 수행 효율 저하를 막기 위해, 대수 추론은 두 가지 조건이 모두 만족된 때에만 행해진다. 앞 절의 명제 추론의 진전이 없을 때(즉, 더 이상의 연산자 적용이 불가능할 때)와, 최근의 대수 추론 이후 작업 메모리의 내용이 변화했을 때이다. 다시 말해서, 명제 추론 전략의 하위 전략으로서만, 그리고 대수 추론에 의해 새로운 사실이 유도될 수 있는 가능성이 있을 때에만 수행됨으로써 불필요한 대수 연산을 최소화한다. 수행 효율 저하를 최소화하기 위한 또 하나의 전략으로는, 대수 추론이 행해질 때마다 목표 가설에 대한 모순 검사를 우선 행하도록 함으로써, 혹시 있을지 모를 모순 가설에 대해 불필요한 고갈적 탐색(exhaustive search)을 방지한다. 예를 들어, "X=25?"이 목표 가설인 경우, 이에 반하는 명제(예를 들어, "X=30")가 작업 메모리에 생성되는 시점부터는 이 가설이 모순임이 판명되어 검증 노력을 중단한다.

대수 추론에 의해 생성되는 명제들에 대해서도 명제 추론에서와 마찬가지로 명제 속성들을 갱신한다. 명제간의 의존 관계를 표현하는 속성도 여기에 포함되는데, 이때의 지지 명제들은 연산자 적용에 따른 것이 아니기 때문에 방정식 해결 근거로서의 지지 명제들을 나타내게 된다. 여기서 대수 추론과 관련하여 생각해야 될 또 하나의 문제가 있다. 특정 수식 명제는, 작업 메모리의 수식 명제들의 상이한 복수개의 조합에 의해 유도될 수 있는 경우가 많다는 문제이다. 예를 들어, 그림 1의 왼

편에 보인 문제 해결이 어느 정도 진행된 상태에서, 오른편 위에 주어진 수식은 아래 상자로 열거된 몇 개의 상이한 조합의 수식 집합에 의해 유도될 수 있다. 그러나 이 가운데 굵은선 상자내의 집단만이 학습 타당성을 가지는, 즉 학습자에게 설명시 사용 가능한 것이라고 할 수 있다. 이러한 학습 타당성을 유지하는 추론을 유도하기 위한 방편으로, 학습기는 수식 명제를 '일반식(general formula)'과 '특정식(specific formula)'의 두 부류로 분리하여 유지한다. 즉, 문제 해결 초기에 주어진 수식들을 특정식으로, 이후 생성된 수식 명제 가운데 특정식의 지지없이 생성된 수식을 일반식으로, 적어도 하나 이상의 특정식에 의해 지지되어 생성된 수식을 특정식으로 정의한다. 그리고 새로운 명제 P가 생성될 때마다 이 정의에 의해 명제 P가 일반식 또는 특정식임을 결정, 이를 명제 P의 속성으로 기록한다. 학습 타당성이 높은 명제 의존 그래프를 얻기 위해서, 대수 추론시에는 작업 메모리의 일반식 속성을 가진 명제들만을 우선 사용하여 유도를 시도하고, 이것이 실패할 경우에만 특정식 속성을 가진 명제들까지도 사용하여 유도를 시도한다. 여기서 소개한 의존 그래프에 의한 논거 관리는 차후 힌트 제공과 설명을 지원하는데 사용될 수 있는 방향으로 계속 연구되고 있는 중이다.

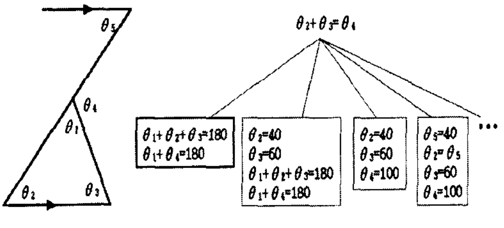


그림 1 수식 "$\theta_2+\theta_3=\theta_4$"의 유도를 가능케 하는 다양한 수식 집단

본 학습기의 방정식 해결기는 독립 모듈로서 어떤 가설에 대한 대수적 해결을 위해 필요한 시점마다 대수 추론부에 의해 호출되어 사용된다. 다양한 전략을 구사하여 계산을 수행하는 방정식 해결기의 입출력 양식은 다음과 같다. 방정식 해결기는 해결 목표와 수식 집단이 전달됨으로써 구동된다. 해결 목표는 "X=30?"과 같은 가설 형식과 "X=?"과 같은 미지수 형식의 두 가지 가운데 하나이다. "X=?" 형식의 해결 목표에 대해서는 계산을 통해 값이 해결되면 "X=<값>"을, 해결되지 않으면 'unknown'을 반환한다. "X=30?" 형식의 해결 목표에 대해서는 이를 일단 "X=?" 형식의 해결 목표로 변환하여 해결을 시도한다. 계산을 통해 값이 해결되면 해결값과 30을 비교하여 'true' 혹은 'contradiction'을 반환하며, 해결되지 않으면 'unknown'을 반환한다. 만약 해결 목표가 "X+Y=Y+X?", "X-X=0"과 같은 항진이면 'tautology'를 반환한다. "X=<값>" 또는 'true'를 반환할 때는 입력 수식들 가운데 해결에 기여한 수식들의 리스트를 함께 반환한다. 방정식 해결기는 학습기의 작업 메모리에 직접 접근하지 않으며, 반환값에 대한 해석 및 처리 역시 대수 추론부에서 이루어 지므로, 독립 모듈로서 작동한다고 할 수 있다.

5.3 미지수 추론

문제 해결 전략의 마지막으로써, 미지수(unknown)에 관한 추론은 앞서의 명제 또는 대수 추론 전략이 모두 실패하여 더 이상 진행이 불가능한 경우, 목표 가설 H에 포함된 미지수를 추출, 이들 각각을 추론의 목표(goal)로 삼도록 추론 방향을 전환 시도하는 것이다. 이 전략은 문제 해결을 위한 추론의 완전성을 위해 필수적인 것이지만, 앞서 두 전략과 같이 어떤 가설을 검증하는데 직접 기여하는 것이 아니라, 탐색의 경로를 수정하여 새로운 방향의 추론을 유도하는 효과를 가진다. 따라서 탐색 공간을 확장한다는 단점을 가지므로 세 가지 전략 중 가장 낮은 우선 순위에서 꼭 필요시에만 구사된다.

이 전략에 의해 생성된 목표 미지수에 대해서 추론은 다음과 같이 진행된다. 목표 미지수 X가 존재하고 작업 메모리에 "X=30" 형식의, 즉 미지수 X만으로 구성된 명제 P(X)가 존재하면 목표가 만족된 것으로 한다. P(X) 형식의 명제가 존재하지 않을 때는 두 가지 경우에 대한 고려를 통해 목표 미지수에 관한 추론을 진행한다. 먼저 경우는, X를 포함한 여러 미지수로 구성된 명제 P(X,…)가 존재할 경우이다. 이때는 X를 제외한 구성 미지수 각각을 다시 목표 미지수로 설정한다. 이는 목표 미지수에 관한 직접적인 추론이 한계에 이르렀지만 구성 미지수에 관해서는 추론이 지속될 수 있는 경우에 유용하게 활용될 수 있는 전략이다. 예를 들어, 목표 미지수가 X이고, 명제 P가 "Z=X-Y"인 경우 구성 미지수 Y, Z을 목표 미지수로 설정함으로써 미지수 Y 및 Z에 관한 추론을 시도하여 그 결과를 X에 관한 추론에 차후 활용하자는 것이다. 다음 경우는, 미지수 X로 우변이 구성된 연산자 O가 존재할 경우이다. 이때는 연산자 O의 우변을 가설로 설정한다. 이는 목표 미지수에 관한 직접적인 추론이 한계에 이르렀을 때, 목표 미지수 해결이 도움을 줄 것으로 보이는 연산자 O의 우변 명제

에 대한 성립 여부를 검증하고자 시도하는 전략이다. 예를 들어, 목표 미지수가 X이고, 연산자 O가 "$P_{1,2,\cdots,N} \Rightarrow X=Y$"인 경우, 우변 "X=Y"를 목표 가설로 설정함으로서 차후 역방향 추론에 의해 "X=Y?"가 검증된다면 이를 미지수 X의 해결에 활용하자는 것이다.

대수를 수반하는 기하 문제 학습의 특성상 위 두 가지 방식의 미지수 추론은 필수적인 것이다. 그러나, 그 가운데 어느 전략이 성공 가능성이 더 높다고 일반적으로 말할 수는 없으므로 상호 독립하여 병행적으로 수행된다.

기존 대부분의 학습기는 대수 처리를 필요로 하지 않는 단순한 증명 유형의 문제들만을 위주로 하여 설계 구현되어 왔다. 그러나 많은 기하 학습 문제는 대수 처리를 필요로 하는 증명 유형, 그리고 특정 미지수의 값을 구하는 해결 유형까지도 포함한다. 본 연구의 의의 가운데 하나는, 이 장에서 제시된 바와 같이, 다양한 유형의 문제에 일반적으로 적용 가능한 해결 전략들을 구현한 데 있다.

6. 학습자 가설 검증

앞서 설명한 추론 전략들에 기반하여 학습기는 학습자에게 제시된 문제의 해결에 관하여 대화식 학습 환경을 제공한다. 학습기는 학습자가 선택한 문제를 로드함과 동시에 내부적으로는 추론기를 구동한다(그림 2).

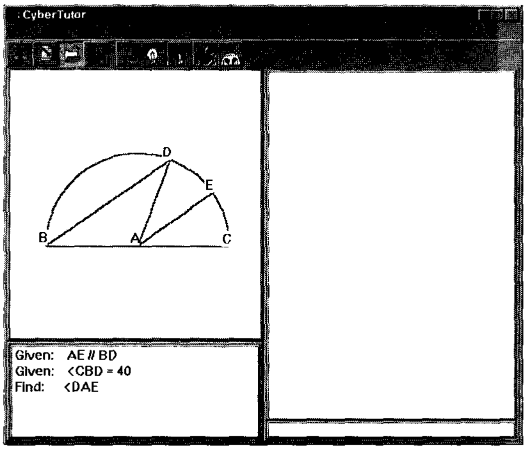


그림 2 문제 학습 초기 화면

이 시점에서부터 학습자와 학습기간의 상호 작용은 학습자로부터의 가설 입력에 대해 학습기가 이를 검증함으로서 반응하는 양식으로 진행된다.

문제 해결의 어느 시점에서나 학습자는 가설, 즉 문제 해결을 위한 단위 명제를 입력할 수 있다. 학습자 가설에 대해 학습기는 추론기를 재구동, 이를 증명 시도한다. 이때 추론기는 앞에서 설명한 문제 해결 규칙들을 적용한다. 추론의 결과 가설이 검증된 경우, 학습자에게 'correct' 메시지를 제시하고, 해당 명제를 학습자가 '알고' 있는 것으로 기록한 후, 다음 입력을 기다린다. 가설의 검증은 여러 가지 추론 경로를 통해 성취된다. 먼저, 가설 입력 시점의 작업 메모리에 가설과 동치인 명제가 이미 존재할 경우 가장 신속하게 검증이 완료된다. 이같은 상황은 이전의 어느 추론기 재구동시에 해당 명제가 생성되었던 경우에 발생한다. 그렇지 않으면, 주어진 가설을 목표로 한 역방향 추론이나 작업 메모리의 명제를 사용한 대수적 해결 또는 이들의 결합에 의해 검증을 완료한다. 이때 미지수 처리를 통한 추론이 수반될 수도 있다. 이 밖에, 학습자 가설이 검증 결과 '항진 명제(tautology)'임이 판명된 경우에는 'correct' 대신 'always true' 메시지를 제시함으로써 주의를 환기한다.

학습자 가설 추적의 가장 큰 특징은 학습자의 수준에 동적으로 적응한다는 점이다. 더딘 학습자가 입력한 가설에 대해서는, 1 단계 또는 적은 단계 수의 추론으로써 검증을 마친다. 반면, 다단계의 논리나 암산에 의한 비약을 자주 수행하는 학습자가 입력한 가설에 대해서는, 중간 단계가 몇 개 또는 모두 생략된 경우가 많으므로 이 때는 여러 단계 수의 추론에 의해 검증을 수행한다. 기존 대부분의 학습기에서는 1 단계의 추론만을 지원함은 물론, 미리 정해진 해결 경로만을 강요하는 체계로 설계됨으로써 학습자 능력에 따른 상이한 추론 양상을 원천적으로 지원할 수 없었다. 이에 반해 본 학습기는 학습자의 다양한 추론 경로와 경로 크기에 적응함으로써 개별 학습 환경의 토대를 마련한다.

만약 학습자 가설에 대한 검증에 실패하면, 학습자에게 'incorrect' 메시지를 제시하고 다음 입력을 대기한다. 이는 추론기가 학습자 가설에 대하여 가능한 모든 추론 전략을 동원하여 검증을 시도한 후에도 실패하였음을 의미한다. 부정확한 가설이 주어질 때마다 추론기가 고갈적인 탐색(exhaustive search)을 거친 후에야 실패함으로써 학습기의 수행 효율을 저하시킬 가능성은 앞서 설명한 '모순 검사(contradiction check)'에 의해 상당히 제거된다. 그림 3은 그림 2 문제에 대한 학습 과정의 일부를 보인 것이다.

본 학습기의 학습자 인터페이스는 학습자와의 상호 작용에 있어서 여러 가지 사용자 친숙한 환경을 제공함으로써 학습의 효과를 높힌다. 기하 영역에서 구현된 학

습자 인터페이스의 경우, 문제 다이아그램의 객체들 가운데 현재 대화의 문맥에 해당하는 객체들을 시각적으로 강조한다거나, 마우스에 의한 선택을 가능케 하는 등이 그 예이다. 이와 함께 인터페이스는 교사 또는 학습자에 의한 문제 입력이나 교사에 의한 공식 입력이 손쉬운 방식에 의해 이루어 질 수 있도록 되어 있다(자세한 내용 [23] 참조).

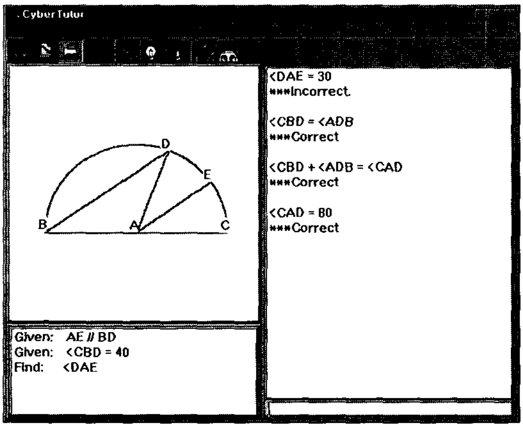


그림 3 학습자 가설 검증

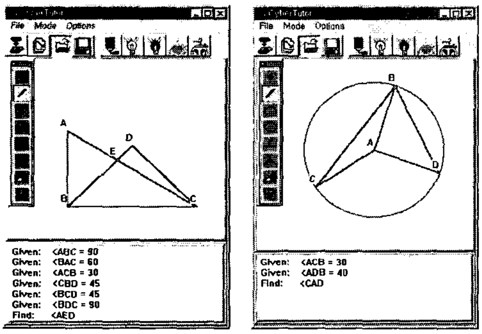


그림 4 본 시스템이 해결 및 교수 가능한 문제 유형

7. 학습 문제 유형

현재 본 학습기의 실험과 평가를 위해 초,중등학교 수준의 기하를 대상으로 문제 베이스를 구축 실험 중에 있다. 문제들은 초,중등학교 교과서나 참고서의 실전 문제들을 원문 그대로 수록하고 있다. 초,중등 수준의 기하 문제로 실험 범위를 제한한 이유는 이들이 대부분 선형 1차 방정식 해결로서 해결될 수 있는 문제로서 본 문제 학습기의 대수 해결 처리 범위 내에 있기 때문이다. 삼각함수나 2차 이상의 방정식 해결을 요하는 문제들을 학습기에서 다룰 수 있기 위해서는 현재의 대수 해결기 부문의 확장을 필요로 할 것이다. 그림 4는 문제 베이스를 예시하는 것으로서, 현재 학습 지원이 가능한 문제들이 제시된 상태의 학습기 화면을 보여 준다.

8. 결론 및 향후 연구

기존 대부분의 문제 학습용 소프트웨어가 미리 저장된 문제의 범위내에서 단답식으로 문제 공간을 향해하는 방식에 의존한 나머지 학습의 흥미를 반감시키고 개인 능력차에 따른 정교한 학습을 지원하기에는 부족함이 많았다. 특히 기하와 같이 암기 위주가 아닌 이해 위주 학습 영역에서의 문제 학습은 일차적으로 문제 풀이의 각 단계를 검증하는 방식을 지원해야 학습의 효과를 기할 수 있다. 이와 같은 필요성에 부응하여 본 연구는 학습자의 실시간 문제 해결을 검증하도록 지원하기 위한 시스템 구성 요소에 관해 정의하고 이를 성취할 수 있는 설계 방안을 제시하였다.

본 연구가 제시하는 문제 학습기 설계 방안의 핵심은 시스템 정보 구성 요소(ontology)의 적절한 선택과 학습과 병행하는 동적 추론 체계에 있다고 할 수 있다. 실시간 문제 해결을 지원하기 위한 정보 구성 요소로서 명제, 가설 및 연산자에 의해 문제 공간을 표현하였고 이들의 생성과 검증을 추론의 주요 목표로 하는 대화식 문제 학습의 메카니즘을 제시하였다. 구체적으로는 명제, 더수 및 미지수의 세 가지 추론 전략을 연계적으로 구사함으로서 대수 처리를 적극적인 방식으로 수용한다. 기존 시스템의 경우 공리 공간 탐색에 기반한 증명 위주의 학습기로만 설계되어 왔으므로 대수 처리를 간과해 왔던 것이 사실이다. 그러나 대부분의 기하 문제 풀이를 위해 대수 처리가 필히 수반된다는 점을 고려할 때, 이는 실용성있는 문제 학습기의 구축에 필수 불가결한 요소라 하겠다. 이런 관점에서 기하 외의 물리, 전자회로 등 대수 처리를 수반하는 과학의 여러 영역의 문제 학습기 설계에도 본 연구가 제안한 설계 원리가 적용될 수 있겠다.

현재 본 연구에서 제안한 추론 체계를 적용하여 학습자의 힌트 요구에 대응한다든가, 주어진 힌트에 대한 설명 등 더욱 다양한 상호 작용을 지원하는 학습 체계를 설계, 개발 중에 있다. 학습자의 수준에 따라 상이한 문제 해결 능력을 가지며, 따라서 다양한 레벨의 힌트와 설명이 요구된다. 이러한 개인 교수식의 학습 환경 구축도 제안된 설계 원리의 자연스러운 확장을 통해 성취되

어 가고 있는 중이다.

참 고 문 헌

[1] Anderson, J., "Acquisition of Proof Skills in Geometry" in [17].

[2] Anderson, J., "The Expert Module" in [18].

[3] Anderson, J., Boyle, F., Corbett, A., Lewis, M., "Cognitive Modeling and Intelligent Tutoring," *Artificial Intelligence*, vol. 42, pp. 7-49, 1990.

[4] Barr, A., Feigenbaum, E., *The Handbook of Artificial Intelligence, vol. 1*, William Kaufmann, 1981.

[5] Barr, A., Feigenbaum, E., *The Handbook of Artificial Intelligence, vol. 2*. William Kaufmann, 1982.

[6] Bobrow, D., *Qualitative Reasoning about Physical Systems*, MIT Press, 1985.

[7] Brown, J., Burton, R., DeKleer, J., "Pedagogical, Natural Language and Knowledge Engineering Techniques in SOPHIE I, II and III" in [20].

[8] Forbus, K., "Qualitative Process Theory" in [6]

[9] Gentner, D., Stevens, A. (eds), *Mental Models*, Lawrence Erlbaum, 1983.

[10] Greeno, J., "Conceptual Entities" in [9].

[11] Koedinger, K., Anderson, J., "Abstract Planning and Perceptual Chunks: Elements of Expertise in Geometry," *Cognitive Science*, vol. 14, pp. 511-550, 1990.

[12] Larkin, J., McDermott, J., Simon, D., Simon, H., "Models of Competence in Solving Physics Problems," *Cognitive Science*, vol. 4, pp. 317-348, 1980.

[13] Larkin, J., Reif, F., Carbonell, J., Gugliotta, A., "FERMI: A Flexible Expert Reasoner with Multi-Domain Inferencing," *Cognitive Science*, vol. 12, pp. 101-138, 1988.

[14] Lawler, R., Yazdani, M. (eds), *Artificial Intelligence and Education, vol. 1*, Ablex, 1987.

[15] Lawson, M., Chinnappan, M., "Generative Activity During Geometry Problem Solving: Comparison of the Performance of High-Achieving and Low-Achieving High School Students," *Cognition and Instruction*, vol. 12, pp. 61-93, 1994.

[16] McDougal, T., Hammond, K., "Representing and Using Procedural Knowledge to Build Geometry Proofs," *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pp. 60-65, 1993.

[17] Michalski, R., Carbonell, J., Mitchell, T. (eds), *Machine Learning*, Morgan Kaufman, 1983.

[18] Polson, M., Richardson, J. (eds), *Foundations of Intelligent Tutoring Systems*, Lawrence Erlbaum, 1988.

[19] Reif, F., "Interpretation of Scientific or Mathematical Concepts: Cognitive Issues and Instructional Implications," *Cognitive Science*, vol. 11, pp. 395-416, 1987.

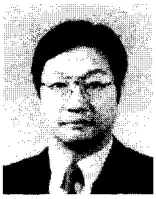
[20] Sleeman, D., Brown, J., *Intelligent Tutoring Systems*, Academic Press, 1982.

[21] Wenger, E. (ed), *Artificial Intelligence and Tutoring Systems*, Morgan Kaufmann, 1987.

[22] White, B., Frederiksen, J., "Qualitative Models and Intelligent Learning Environment" in [14].

[23] 국형준, "효과적 문제 해결 학습을 위한 지능형 기하 교수 시스템", 한국정보과학회 논문지(B), vol. 25, no. 7, pp. 1090-1100, 1998.

[24] URL1, http://www.ghg.net/clips/CLIPS.html, "Clips: A Tool for Building Expert Systems".



국 형 준

1979년 서울대학교 공과대학 학사. 1983년 Univ. of South Carolina 전산과학 석사. 1989년 Univ. of Texas at Austin 전산과학 박사, Post-Doc. 현재 세종대학교 컴퓨터공학과 부교수. 관심분야는 인공 지능, 전문가 시스템, 지능형 교수시스템, 지능형 에이전트.