

## 협력학습을 위한 진단과 스케줄링 에이전트 시스템

한선관\* · 김세형\*\* · 조근식\*\*\*

### Diagnosis and Scheduling Agent Systems for Collaborative Learning

Sun-Gwan Han\* · Se-Hyoung Kim\*\* · Geun-Sik Jo\*\*\*

#### 요 약

본 연구는 웹 상에서 원격 협력 학습을 위한 수준별 협력 학습자 진단 및 스케줄링 에이전트의 설계와 구현에 관한 연구이다. 원격 협력 학습은 동일한 학습내용에 흥미를 갖는 아동이 동시에 학습할 수 있는 환경이 필요하며, 학습자의 지식 또한 비슷한 수준이어야 효과적인 협력학습을 할 수 있다. 분산 환경의 이질적인 학습자를 모으기 위해서는 좀 더 자율적이고 지능적인 시스템이 필요하며, 학습자에 대한 지식을 표현하는 학습자 모델이 요구된다. 이를 위해 에이전트 시스템이 적절하게 사용될 수 있으며 학습자의 수준을 판단하기 위한 진단 에이전트와 협력학습이 가능한 여러 명의 학습자들을 알맞은 시간과 서버에 연결하는 스케줄링 에이전트를 웹기반 지능형 교수 시스템에 접목하였다. 학습자 수준을 진단하는 진단 에이전트는 확신도를 높이기 위해 3-모수 로지스틱 확신 공식과 시간 가중치 확신인자 공식을 적용하여 신뢰도를 높였다. 또한 협력학습의 스케줄링을 위해 다양한 제약 조건들의 최적해를 구하기 위해 제약 만족 문제(CSP)로 스케줄링 에이전트를 모델링 하였다. 본 연구에서 설계 구현한 협력학습자 진단 및 스케줄링 에이전트의 효율성을 살펴보기 위해 여러 명의 학습자를 대상으로 실험하였다. 실험을 통해 각 학습자의 지식 수준 진단과 다수의 학습자가 적절한 협력학습을 하기 위한 스케줄링이 매우 효율적으로 이루어짐을 볼 수 있었다.

Key Words : Intelligent Tutoring System, Collaborative learning, Dynamic Scheduling Constraint Satisfaction Problem, Intelligent Agent System

\* 인하대학교 전자계산공학과

\*\* 인하대학교 전자계산공학과

\*\*\* 인하대학교 전자계산공학과 부교수

## 1. 서 론

오늘날 인터넷에 접속할 수 있는 가장 쉽고, 가장 인기 있는 방법인 월드 와이드 웹의 등장과 함께 인터넷은 가장 중요한 교수-학습 도구로서 교사들에게 인식되고 있으며, 웹을 이용한 새로운 교수 모형에 대한 시각이 나타나고 있다. 새롭게 출현하고 있는 이 교수 모형을 웹기반 학습(Web Based Instruction, WBI)이라고 부르고 있는데 이는 특정한 방법으로써 학습자의 지식이나 능력을 육성하기 위한 의도적인 상호작용을 웹을 통해 전달하는 활동이라고 정의 내릴 수 있다(David H. Jonassen, 1996).

그러나 현재 운용되고 있는 교수 시스템들은 대부분 일방적인 자료의 전달을 통해 주어진 과제의 암기가 학습목표가 되고 있어 학습자의 심층적 사고 과정에 개입하여 적절한 자극을 줄 수 없으며, 학습자가 이미 알고 있는 내용에 새로이 학습한 정보를 의미 있게 연결시키기 어려워 진정한 의미의 학습이나 심도 있는 이해를 가져오기 어렵다. 이를 극복하기 위한 방법으로 지능형 교수 시스템 (Intelligent Tutoring System, ITS)의 필요성이 대두되었다(Badrul, 1997).

웹 상에서의 ITS는 각각의 모듈이 분산화 될 수 있으며, 각 모듈은 지능적인 에이전트 Systems에 의하여 상호 통신과 협력적인 작업을 지능적이며 자율적으로 처리할 수 있다.

웹 상에서의 협력학습은 공통의 주제를 학습하려는 다중의 학습자 클라이언트와 협력학습이 가능한 지능형 교수 학습 모듈을 가진 서버로 구성되며 이를 자율적으로 중개해 줄 수 있는 중개 에이전트가 필요하다. 또한 원활하고 효율적인 협력학습을 위해서는 협력학습에 적합한 교수학습전략과 원격 교육 시스템이 필요하지만

지금 구축되고 있는 대부분의 웹기반 학습은 이를 뒷받침 해주지 못하고 있는 실정이다. 또한 협력 학습이 효율적으로 이루어지기 위해서는 적합한 수준의 학습자를 선정하여 교수학습 전략에 알맞게 그룹을 형성하는 작업이 가장 먼저 선행되어야 한다(Marry A. Nixon, 1999).

따라서 본 연구는 효율적인 협력학습을 위한 웹기반의 ITS 기본 구조를 설계하고, 다양한 교수 학습 전략을 탐색한다. 그리고 협력 학습 주제의 선행 지식 수준을 문항 반응 이론을 적용하여 학습자의 지식 수준을 진단하는 진단 에이전트를 지식 기반 구조로 설계하며, 다양한 학습자들을 선별하여 학습자가 원하는 학습 시간에 원격으로 협력할 수 있는 제약 기반 스케줄링 에이전트를 설계, 구현하였다. 또한 교육용 EKQML을 정의하여 각 에이전트의 정보와 지식을 통신할 수 있도록 하였다.

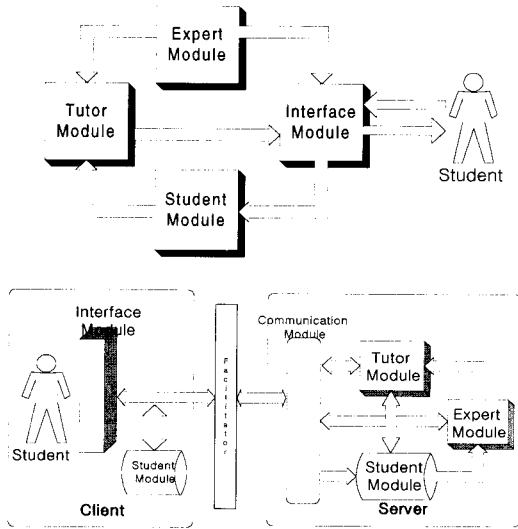
이러한 연구를 위해 웹 상의 여러 학습자와 서버의 교수학습 모듈간에 협력학습을 할 수 있도록 기본 협력 학습 시스템과 이 구조에 따르는 진단 에이전트와 협력 학습 스케줄링 에이전트를 구현하여 학습자에게 적용해 봄으로써 그 효율성을 입증하도록 하였다.

## 2. 이론적 고찰

### 2.1 웹 기반 ITS

기존 CAI의 정적이고 제한적인 단점을 극복하여 지식 기반의 지능적인 기본 ITS의 구조를 살펴보면 [그림2-1] 위의 그림과 같다(Marry et al, 1998).

전통적인 ITS의 모형은 한 대의 컴퓨터에 모든 모듈이 설계되어 있어 단일 사용자를 대상으



(그림 1) ITS와 웹기반 ITS의 구조

로 지능적으로 학습을 도와주었다. 이러한 구조의 단점은 다양한 학습자의 지식을 공유하거나 이용하지 못한다는 점이다.

그러나 ITS는 인터넷이라는 웹의 발달로 지식의 공유가 확산되며 ITS도 [그림1]의 아래 그림과 같이 분산 환경에 맞게 발전하게 되었다. 웹기반 ITS의 구조를 보면 일반적으로 학습자 모듈과 인터페이스 모듈이 클라이언트에 위치하고 교수학습 모듈과 전문가 모듈은 서버에 위치하게 되어 지식을 공유할 수 있게 된다. 이러한 분산환경의 특성으로 말미암아 Client-Server간의 통신을 위해 통신 에이전트와 이를 적절하게 조절을 해 줄 수 있는 조정 에이전트(Facilitator)가 포함된 새로운 구조의 지능형 시스템이 필요하게 되었다(Marry et al, 1998).

## 2.2 지능형 에이전트

에이전트 시스템은 사용자의 컴퓨터 업무를 도와주거나 대신해 주는 지적인 특성을 갖는 프

로그램이라고 정의할 수 있다. 다시 말해서 에이전트는 지식 베이스와 추론 기능을 가지며 사용자, 자원 또는 다른 에이전트와 정보 교환과 통신을 통해 문제 해결을 도모하는 소프트웨어로서 이들이 가지는 고유의 통신언어와 규약을 통해 분산된 이질의 시스템들 간의 지식 공유를 통한 CSCW (Computer Supported Software Work)를 구현하기 위한 것이다(Barbara et al, 1995).

에이전트의 특성들을 살펴보면 사회성(Social ability), 자율성(autonomy), 반응성(Reactivity), 선행성(Pro-activeness), 협동성(Cooperative relationship) 등을 가지고 있으며, 일을 어떻게 수행하는가에 따라 다음의 3가지로 나눌 수 있다(Nicholas R. Jennings et al, 1995).

- Single 에이전트 : 단일 에이전트로 고유 분야에서 사용자에게 서비스를 제공
- Cooperating 에이전트 : 몇 개의 에이전트 간에 상호 협력을 통하여 공통된 목표를 달성하기 위해 서비스를 제공
- 에이전트 사회: Cooperating 에이전트보다 더 광의의 개념이며 이동 에이전트가 이에 속한다.

다중 에이전트간의 대화는 정해진 언어 규약에 의해 메시지를 주고받게 된다. 서로 다른 언어로 개발된 프로그램간의 공통의 통신규약은 ACL (에이전트 Communication Language)을 사용하게 되는데 가장 대표적인 것은 미국방성인 DARPA에서 개발한 KQML (Knowledge Query Manipulation Language)을 주로 사용한다. KQML은 화행 이론(Speech-Act Theory)에 기반을 두고 First Order Logic 형태를 갖는 메시지에 대한 문법적 구조와 어휘를 제공할 뿐만 아니라 통신 규약을 포함하고 있다(Finin & Labrou et al,

1995, Gerherd, 1999). 또한 에이전트와 프로그램이 정보나 지식을 교류할 수 있는 기반 구조를 제공하고 있다.

### 2.3 제약 만족과 동적 자원 스케줄링 문제

학습자의 학습 시간 스케줄링은 선호하는 학습 시간, 기존의 개인 일정, 서버의 부담을 덜어주기 위한 학습 서버의 일정 등의 여러 가지 제약 조건을 만족하면서 효율적인 학습을 할 수 있도록 스케줄링 되어야 한다.

학습자들은 누구나 능률적인 학습하기 위해 좋아하는 시간이 있다. 또한 자신의 기존 일정이 있으므로 어느 특정의 시간을 이용하여 학습하기를 바란다. 그러나 학습자들의 선호도는 각자 다르며 기존의 일정 또한 누구나 다르다. 또 협력 학습이 실시된다고 하더라도 서버의 이용자가 많을 경우 전송 속도가 현저히 떨어져 효율적인 학습을 기대하기 어렵다.

다양한 학습자가 존재할 경우 이러한 여러 가지 제약 조건들을 만족하는 시간을 탐색하는데는 많은 시간과 노력이 필요하며 효율적인 방법으로 수행되지 않으면 매우 느린 결과를 보여주거나 학습이 불가능할 것이다. 또한 한 명의 학습자라도 학습을 취소하거나 학습서버의 문제로 인하여 전체의 스케줄링이 바뀔 경우 그 영향이 모든 학습자들에게 미치기 때문에 에이전트간에 동적인 스케줄링 기법(Dynamic Resources Scheduling)을 적절히 적용해야 한다.

따라서 협력 학습 시간을 스케줄링 하는 문제는 다양한 학습자와 서버의 시간 자원의 제약 조건들을 만족하는 해를 구하는 동적 자원 스케줄링 문제에 속한다.

이러한 제약 조건을 효율적으로 해결하기 위한 방법으로 제약 만족 문제(Constraint Satisfaction

Problem, CSP)가 있다.

CSP기법은 NP계열의 이산 복합 문제들을 효과적으로 해결하는 방법으로  $m$ 개의 변수(Variables)와 그 변수가 가질 수 있는  $n$ 개의 유한 이산 도메인 값(Values) 그리고 변수들 사이의 제약 조건(Constraints)으로 구성이 되며, 각 변수가 가지는 유한 이산 도메인 내에서 문제에 주어진 모든 제약 조건들을 만족하는 해를 찾는 문제로 정의된다(B. A. Nadel, 1989).

이러한 CSP문제를 효율적으로 해결하기 위해서는 각각의 도메인 변수와 할당되는 값 그리고 제약 조건을 문제 해결에 알맞게 지식을 정형화해야만 한다. 정형화된 지식 표현 방법은 제약 관계의 동적인 스케줄링 문제에서 적합한 해를 찾기 위해 많은 도움을 준다. 정형화된 문제는 제약 조건 네트워크로 구성할 수 있으며 변수는 노드로 도메인은 노드 안의 요소로 표현되며, 제약 조건은 노드들간의 아크로 표현될 수 있다.

많은 학습자가 더 나은 학습을 하기 위해 자기에게 알맞은 학습자를 선택하여 학습을 하려고 시도할 것이다. 하지만 학습자 자신은 일정을 따로 관리하기 때문에 학습 시간 스케줄링에 필요한 정보나 지식은 자연적으로 분산되어 있으며 어느 누구도 학습 전체적인 활동에 대해 모르고 있다(E. H. Durfee et al., 1992). 이것은 분산 시스템의 일반적인 특성이라 할 수 있으며 이로 인해 학습자간의 학습 시간에 충돌이 발생할 수 있다.

따라서 이 충돌을 해결하기 위해서 스케줄링 에이전트는 지식 기반의 스케줄링 방법과 적합한 학습자를 찾기 위한 전략을 채택해야 한다. 이를 만족하는 결과를 탐색하지 못했을 경우 제약조건들을 조금씩 완화하여 탄력 있게 새로운 결과를 적절하게 제공해야 한다.

### 3. 협력학습을 위한 진단과 스케줄링 에이전트 시스템의 설계

#### 3.1 기본 구조

본 시스템의 구성도는 [그림 2]와 같이 3-레이어로 구성되었으며 학습자가 서버와 상호 작용할 수 있는 인터페이스 모듈은 각각의 클라이언트 영역에 위치하고 있으며 나머지 모듈들은 각각의 학습을 가지고 있는 서버에 위치하고 있다. 멀티 에이전트의 기본 구조에서 볼 수 있는 것과 같이 Facilitator라는 중개 에이전트가 서버와 클라이언트 사이에 존재한다.

또한 학습자들을 진단하고 협력학습을 중개해주는 중개 에이전트 서버가 따로 존재하여 학습

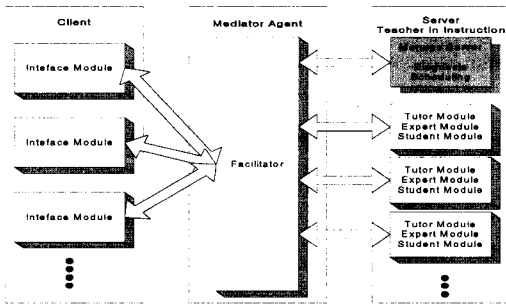
자와 학습 시스템을 관리해 준다.

이렇게 학습자와 학습서버를 따로 관리할 때의 이점은 학습 서버의 고유 기능을 그대로 유지하여 기본 교수-학습을 하도록 하며 기타 다른 기능들은 관리를 따로 하여 서버의 과부하를 막고 다양한 학습자들의 자료를 통하여 일관된 학습자 모델의 생성이나 관리의 효율성이 있기 때문이다.

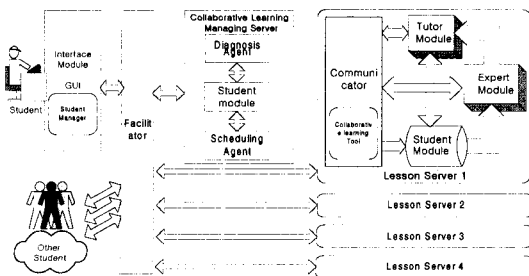
이러한 기본적인 네트워크 체계를 기반으로 하여 구성되어지는 협력 학습 시스템의 세부 모델은 [그림 3]과 같다. 각 클라이언트에 인터페이스 모듈과 학습자 모듈을 두고 서버에 교수학습 모듈과 전문가 모듈을 구성한다.

진단 에이전트는 인터페이스 모듈을 통해 입력되는 학습자의 기초 사항과 요구 사항을 가지고 학습에 맞는 진단 검사를 실시하고 학습자의 학습 수준 및 지식을 추론한다. 그 다음에 학습자의 요구 사항과 추론된 지식수준을 스케줄링 에이전트에게 전달한다.

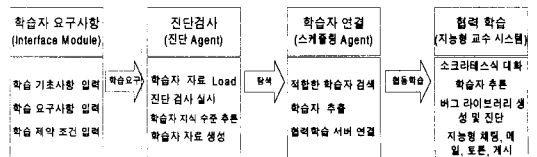
스케줄링 에이전트는 전달된 정보를 가지고 Facilitator에 등록되어 있는 다른 학습자를 웹 상에서 검색하게 된다. 비슷한 지식 수준을 가지고 같은 주제를 학습하고자 하는 다른 학습자를 찾게 되면 협력 학습을 제의하게 되고 이를 수락할 경우 학습하고자하는 시간에 공통영역의 협력학습을 시작하게 된다. 전체적인 협력학습의 과정을 단일 사용자를 기준으로 간단한 시나리오로 구성된 설명이 [그림 4]에 나타나 있다.



[그림 2] 협력 시스템의 기본 구성도



[그림 3] 협력 학습 시스템의 세부 모델



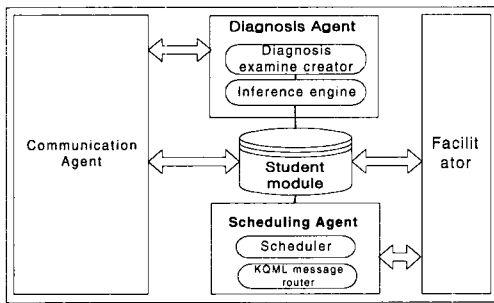
[그림 4] 협력학습 시나리오

### 3.2 진단 에이전트

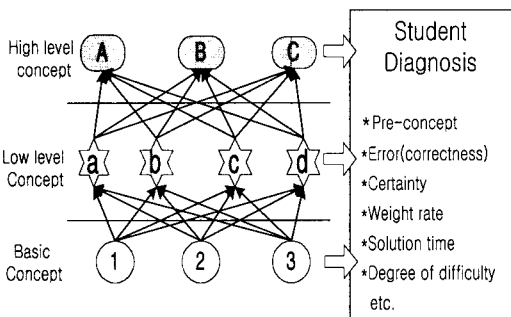
진단 에이전트는 학습자가 요구 사항을 토대로 학습자를 진단을 하며 그 결과를 토대로 학습자 모델을 생성하게 된다. 이러한 학습자 모델은 다양한 학습의 방법에 사용될 수 있으며, 비슷한 학습자가 동일한 학습을 하고자 할 때 진단이 없이도 학습자의 상태를 추론할 수 있는 모델로 사용할 수 있다.

물론 학습자의 기초 사항과 입력 사항을 토대로 추론을 한다. 이에 대한 결과 값은 학습자 모델에 다시 저장이 되고 이 값에 의해 스케줄링 에이전트가 협력학습에 참여할 각 학습자들을 모집하게 된다.

[그림 5]는 진단 에이전트의 구조를 나타낸 것



[그림 5] 진단 에이전트의 구조



[그림 6] 수준별 문항 진단 과정 모형

이다. 진단 에이전트는 내부에 진단 검사 생성기와 추론 엔진을 포함하고 있으며 학습자 모델과 링크 되어 학습자 진단을 지속적으로 유지할 수 있다.

진단의 과정은 [그림 6]에서 보는 바와 같이 수준별 진단 문제에서 각 문제간의 관계 및 정오답 확인, 해결 시간, 난이도, 가중치 등이 규칙 기반 생성 시스템(Rule based Production System)의 형태로 표현되어 추론된다.

<Rule Base의 예>

IF 1 = known and a = known  
THEN A = known

IF 2,3 = unknown and b = known  
THEN B = unknown

진단 검사를 이용한 추론 방법은 규칙 기반 시스템을 이용한다. 진단 문제는 3단계의 수준으로 나누고, 여기에 가중치(Weight rate)와 해결 시간(Solving time)을 적용하여 문제 해결에 따른 학습자의 지식 수준을 추론하게 된다. 위에 예를 보인 것과 같이, 하위 수준을 모르는 경우 상위 수준 문제의 정답을 맞추었을지라도 학습자가 상위 수준 문제를 실제로 알고 있다고 볼 수 없다. 또한 학습자가 추측에 의해서 문제를 해결할 경우를 위하여 Birnbaum의 3-모수 로지스틱 공식을 적용한 확신도(Certainty Factor) 공식을 이용하여 문제에 대한 신뢰도를 높일 수가 있다. 다음 [공식 1]은 가중치와 문제 해결 시간에 따라서 학습자의 문제 해결 능력을 확인하는 공식으로 본 연구에서 제안한 Temporal Weight 확신인자 공식이다.

각 문제에서 맞힐 확률(가중치)은 위의 식(a)와 같이 3-모수 로지스틱모형을 이용하여 구할 수 있으며, 이 식을 확신 인자 공식(b)에 적용하면

다음과 같다.

$$(a) \quad W_i(\theta) = c + (1-c) \frac{1}{1 + e^{-a(\theta - b)}}$$

$W$  : 문제의 가중치

$\theta$  : 학습자의 수준

$e$  : 지수 상수 (2.718)

$a$  : 문항 난이도 모수

$b$  : 문항 변별도 모수

$c$  : 문항 추측도 모수

$$(b) \quad CF(X) = \sum_{i=1}^n \left( \frac{\sum_{l=1}^L (P_{li} \times W_i \times T_{li})}{A} \right)$$

$X$  : 확신 인자 값 ( $X < 1$ )

$A$  : 모든 문제의 집합

$l$  : 문제의 레벨 (High Middle, Low)

$i$  : 각 레벨의 문제의 수

$P$  : 문제의 정오답 (0 또는 1)

$W$  : 각 문제의 가중치 (전체 합=1)

$T$  : 각 문제의 해결 시간

[공식 1] Temporal Weight 확신인자 공식

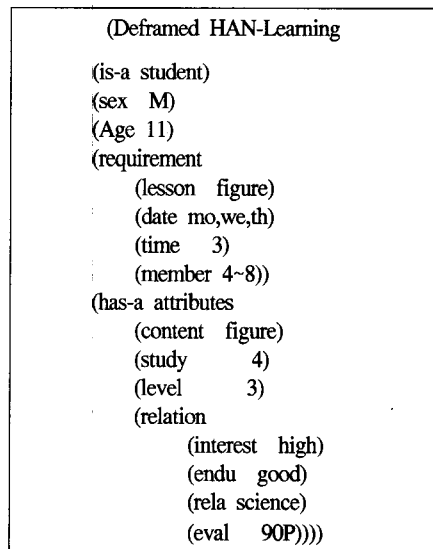
에이전트의 진단 평가 내용은 학습 영역 설정에 있어 중요한 자료로 활용되며 협력 학습자들을 매칭 시키는 역할을 한다. 또한 학습자 모델을 구성하는데 중요한 역할을 수행한다.

학습자 모델을 구분하기 위한 기초자료로 학습자의 기초 입력 자료와 요구사항을 입력받으며 그 내용은 [표 1]과 같다.

학습자 모델은 프레임의 지식 형태로 표현될 수 있으며, 3부분으로 구분하여 관리한다. 학습자의 기초적인 입력 사항과 학습에 필요한 학습자의 요구 사항(제약 조건), 그리고 진단과 학습 결과에 대한 내용들로 구성되어 이를 토대로 학습자의 지식 수준을 추론하고 학습 상황을 파악한다.

[표 1] 학습자 모델

구분	항목	예	
기초 입력 사항	성명	Han Sun Gwan	
	성별	man	
	나이	11	
	학년	5	
	언어	Korean	
요구 사항 제약 조건	성명(ID)	학습자 고유 ID -Han	
	과목	수학	
	주제	도형	
	학습 가능일	월, 수, 목, 금	
	학습가능 시간	PM 7-10시 AM 2-5시	
	주간 학습시간	3시간	
	학습 인원	4-8명	
학습자 수준 자료	학습한 내용	도형	
	학습한 시간	3회 4시간	
	각 지식 수준	평면도형 이해(CF 0.9)	
	각 학습 영역 관계	흥미도:	높음
		지속력:	좋음
		흥미과목:	과학
학습태도:		매우 좋음	
평가결과:	총괄(90점)		



[그림 7] 학습자 지식의 프레임

[그림 7]은 위에 제시한 학습자 테이블을 프레임의 지식으로 표현한 것이며, 이러한 프레임을

통해서 학습자와 관련된 다양한 자료들을 분석하고 구축, 관리를 더욱 편리하게 해줄 수 있다. 이 학습자 모듈의 내용은 인터페이스 모듈을 통해 입력이 되고 진단 평가와 학습 후에 학습에 대한 결과가 학습자 모듈에 저장되어 다음의 학습과 스케줄링에 이용된다. 또한 학습 과정 중의 학습자의 태도 및 평가 내용 등이 저장된다.

### 3.3 스케줄링 에이전트

스케줄링 에이전트는 학습자의 요구사항들(계약 조건들)과 진단 에이전트에 의해 평가받은 결과에 의해 전달받은 학습자의 지식 수준을 가지고 협력학습에 적합한 학습자를 검색하고 모집하는 에이전트이다.

학습자 모듈과 진단 에이전트와 통신하는 기능은 KQML의 메시지에 의해서 가능하다. KQML에 의해 전달된 정보들은 학습자간의 시간이나 학습하고자하는 내용, 지식 수준에 따라 필터링되고, 중재하며 연결시키는 기능이 필요하다. 특히 동일 영역의 학습을 같은 시간 내에 하기 위해서는 스케줄링이 필요한데 협력학습을 할 인원이 많아지게 되면 복잡한 계산이 필요하게 된다.

효율적인 스케줄링을 위해 CSP로 정형화하기 위한 학습자의 시간의 동적 자원 스케줄링 문제는 다음과 같은 제약 조건과 우선 순위가 있다.

스케줄링을 위한 지식 표현(Knowledge representation)

$CL = \{LI, NS, LL, LT, LS\}$

- 협력 학습(collaborative learning, CL)
- 학습 주제(learning item, LI)
- 학습 시간(learning time, LT)
- 학습 서버(learning server, LS)
- 학습 인원(number of student, NS)

- 학습 수준(learning level, LL)

계약 조건(Constraints)

- 1차 학습 시간이 2차 학습 시간보다 먼저 할당되어야 한다. ( $LT1 < LT2$ )
- 학습자 개개인은 각자의 일정이 따로 있으며, 동일시간에 여러 개의 일정이 겹칠 수 없다. ( $LT1 \neq LT2$ )
- 각 협력학습에는 적절한 수준의 학습자가 참여해야 한다. 같은 수준, 다른 수준 등등 ( $LT1 \in \{h, m, r\}$ ,  $highlevel = midlevel = lowlevel$   
 $LT2 \in \{h, r\}$   $h=1, r>1$   
 $LT3 \in \{h\}$  or  $LT4 \in \{m\}$  or  $LT5 \in \{r\}$ )
- 각 협력학습의 인원은 제한되어 있다. ( $CL1 = \{CS < 5\}$   $CL2 = \{CS < 7\}$  ... )
- 각 학습 서버도 따로 존재하며 일정이 있다. ( $LS1 = \{CL1\}$   $LS2 = \{CL2\}$  ...)

우선 순위(Preferences)

- 학습자들은 자신의 먼저 학습을 선택한 시간에 학습하길 좋아한다. ( $CS1 = \{LT1 \succ LT3\}$   $CS2 = \{LT2 \succ LT3\}$ .)
- 학습시간은 변경이 되지 않는 것이 좋다. ( $LT1, LT2 \succ change$ )
- 자신이 좋아하는 학습 과목과 내용이 있다. ( $CS1 = \{CL1 \succ CL2\}$   $CS2 = \{CL4 \succ CL2\}$ ...)
- 학습 서버는 가급적 성능이 빠른 것이 좋다. ( $CL1 = LS1$ )  $\succ$  ( $CL1 = LS2$ )
- 협력 학습자가 적정 인원이어야 좋다. ( $5 \leq CL1cs \leq 7$ )

이러한 정형화된 지식의 표현을 가지고 JCL(Java Constraint Library)로 해결하기 위한 CSP 기술 언어(Constraint Satisfaction Problem Description Language, CSPDL)로 표현하면 다음과 같다.



```

Network
  Name: Schedule Network
  Author: Han
Variables
  v1: d1(CL) // collaborative learning
  v2: d2(LD) // learning date
  v3: d3(LT) // learning time
  v4: d4(LS) // learning server
  v5: d5(CS) // number of student
  v6: d6(SL) // student level
Domains
  d1: geo, number, set, cal, ...
  d2: mon, tue, wed, thu, fri, sat, sun
  d3: 9,10,11,12,13,14,15,16,17,18,19,20
  d4: bad, common, good
  d5: few, mid, many
  d6: high, mid, low, homo, hetero
Constraints
  v1: (geo<number),(geo<set), ...
  v1, v5:(geo, few), (number, common),
    (cal, many),(set, few),...
  v1, v6:(geo, high), (number, low),
    (set, high), (cal, mid), ...

:language EJESS
:ontology Collaborative-learning
:content (lesson_OK
:s_subject student1 sub?
:s_date student1 date?
:s_time student1 time?
:s_level student1 level?))
(tell
:sender student1-agent
:receiver scheduling-agent
:in-reply-to sch-agent9909110523
:language EJESS
:ontology Collaborative-learning
:content (lesson_OK
:s_accept student1 1001001
:s_subject student1 figure
:s_date student1 Mon
:s_time student1 P10,P11,A3,A6
:s_level student1 high-0.9))

(ask
:sender scheduling-agent
:receiver student1-agent
:reply-with sch-agent9909110523

```

스케줄링을 위한 제약 조건은 모두 17개가 사용되었으며 우선 순위를 위한 조건은 9개를 사용하였다.

다음의 KQML 예제는 스케줄링을 위한 학습자의 지식 수준과 협력학습 내용과 시간을 요구하고 전달하는 예이다.

이렇게 KQML을 통해 각각의 학습자 에이전트에게서 전달받은 정보를 가지고 스케줄러가 교수학습 전략에 맞는 학습자들을 협력학습에 맞도록 스케줄링을 한다.

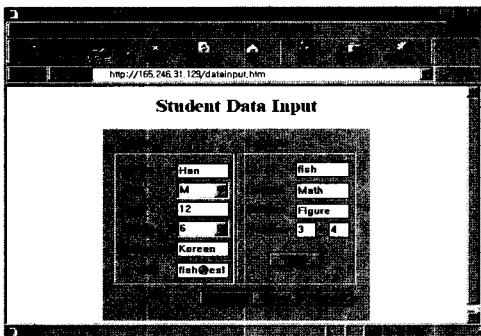
우선 제약 조건에 따라 학습 가능한 학습자들을 찾은 후에 각각의 학습자의 시간을 파악한다. 서버는 사용자 수에 따라 Full, Common, Empty로 나누고 학습자의 시간은 각각 일정에 따라 Yes, No, Blank로 나눈다. Server와 각각의 학습자들의 시간을 탐색하며 적합하지 않은 시간(No or Full)을 제거해 간다. 학습 시간이 같은 경우의 학습자들은 적절한 시간을 할당받아 학습을 하게 되며, 적합하지 않은 학습자들을 제약 조건을 완화하여(Common or Blank) 적절한 시간을 할당받는다.

동일한 시간에 학습자가 많을 경우 협력 학습에 무리가 있으므로 평가 함수에 의해 학습자를 우선 할당할 수 있다. 평가 함수는 진단 에이전트에 의해 전달된 학습자의 학습 태도를 가지고 평가할 수 있다. 각각의 에이전트가 경쟁과 타협을 통해 가장 좋은 학습자를 선택하게 되는 것이다.

#### 4. 시스템의 구현 및 실험 고찰

본 연구에서 설계하고 구현한 원격 협력 학습 ITS 서버는 운영체제를 Windows NT환경에서 JAVA 기반의 애플릿을 이용하여 인터페이스와 통신 서비스 부분을 구현하였고, MS-SQL을 사용하여 지식 베이스를 구축하였다.

진단 문제와 인터페이스 부분의 제작은 교육용 저작언어인 PASS2000을 이용하여 웹브라우저에 플러그인 하였다. 추론 엔진은 JESS(Java Expert System Shell)를 사용하여 지식의 생성과 패턴 매칭에 사용하였으며 멀티 에이전트간의 Message를 교환하기 위해 JKQML을 사용하였다. 학습자의 동적인 스케줄링 문제를 해결하기 위하여 JAVA 기반의 제약 문제 해결기인 JCL(Java Constraint Library)를 이용하였다.



(그림 8) 기초 사항 입력



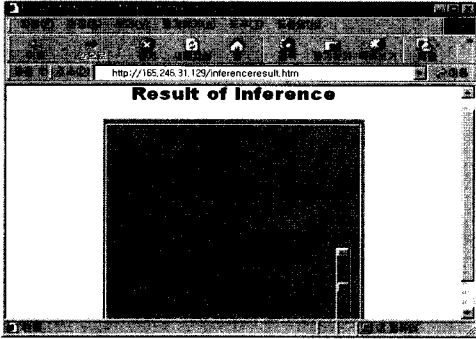
(그림 9) 진단 검사

학습자 클라이언트들과 다양한 학습 서버 및 학습 관리 서버들을 중개해 주는 중개 에이전트로 Facilitator를 설치하여 다양한 학습자와 서버들을 관리한다. 인터페이스 모듈을 통해 입력되는 사용자 기초자료와 요구사항을 보면 [그림 8]과 같다. 인터페이스 모듈을 통해 입력 값들은 진단 검사에 사용되며 협동학습시 학습자 모델에 적용이 된다.

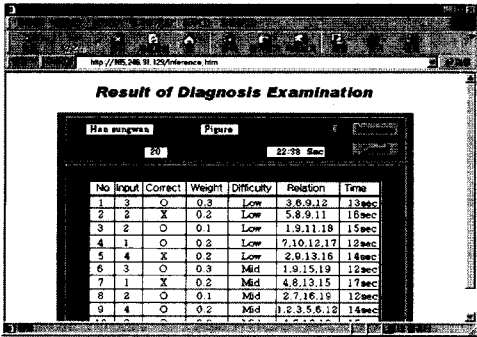
진단검사는 Pass 2000 프로그램을 이용하여 [그림 9]와 같이 도형 학습에 관한 표준화된 수준별 문제 20문항을 제시하여 학습자의 지식수준을 검사하였다. 진단 문제의 위계성 및 문제난이도는 Van Hiele의 도형 위계학습에 관한 내용에서 추출하였다.

각각의 문제들은 서로 관련되어 있으며, 이러한 진단 검사의 신뢰도를 위하여 20문항을 제시하고 실제 학습자의 수준을 검사하는 문제는 10문항으로 두어 학습자의 추측 요인과 일관성에 관한 추론을 하도록 하였다. 진단 검사 결과를 추론엔진(Inference Engine)을 통하여 추론한 예제 화면을 [그림 10]에 나타내고 있다.

본 연구에서 구현한 시스템으로 협력 학습 시간을 스케줄링하기 위해 [그림 11]과 같이 입력

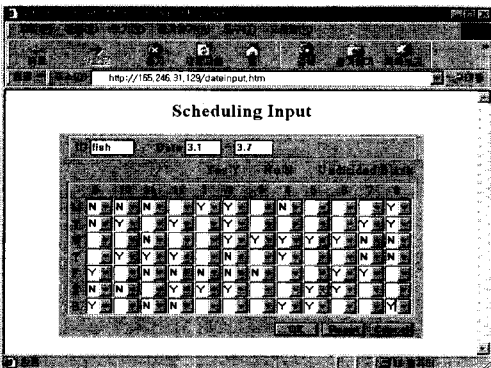


(a)



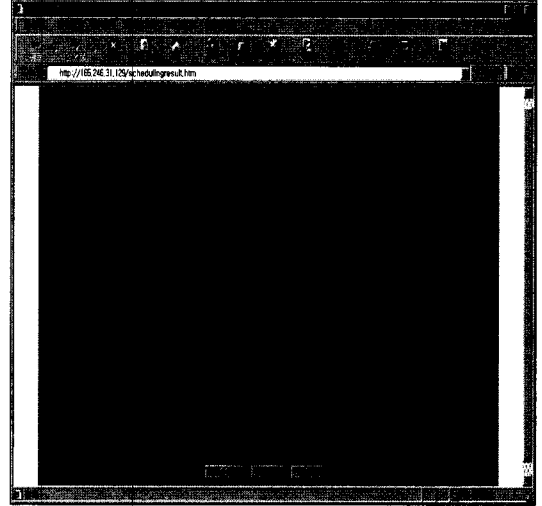
(b)

(그림 10) 진단 검사의 추론 결과



(그림 11) 학습 가능시간 입력

할 수 있다. 학습자는 학습할 수 있는 요일을선택하며, 각각의 시간에 자기가 학습 가능한 선호도를 자신의 일정에 맞게 입력하여 스케줄링 한다.



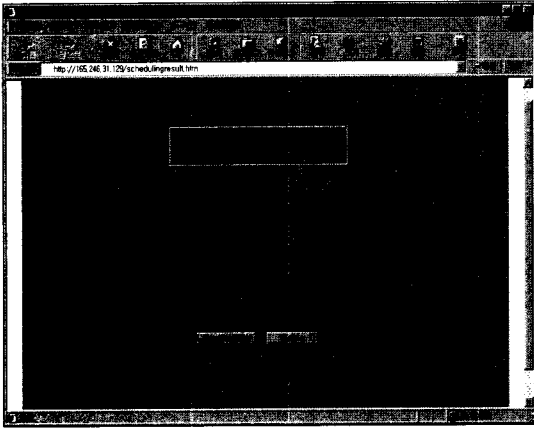
(그림 12) 시간 테이블 간트 차트

우선 학습할 서버의 일정을 확인 후 가능 여부를 제시하고, 여러 학습자의 스케줄을 일렬로 제시한다. 서버의 bad과 학습자의 No를 선택한 시간을 삭제하고 서버의 Empty부분과 학습자가 Yes를 선택한 시간을 추출한다. 학습자 수가 너무 많이 한 서버로 모일 경우, 적당한 수로 나누어 학습 시간을 할당하는데 이때 학습자 모델의 평가 함수를 이용하여 최적의 학습자끼리 연결시켜준다.

만약 학습자가 적을 경우 제약 조건을 완화하여 서버의 Common과 학습자의 Yes 또는 Blank를 선택한 시간을 할당한다.

실험은 학습자 22명을 대상으로 실험하였으며, 각 학습자는 진단에이전트에 의해 진단 평가를 받고 학습자의 지식 수준을 평가하게 된다. 다음 자신의 시간을 입력하여 협력 학습 가능 시간의 결과를 받아 보게 하였다.

[그림 12]의 시간 테이블 간트(Gant) 차트는 각 학습자들의 지식 수준과 학습 요구 시간을 나열한 것이고 스케줄링의 결과를 보면 지식 수준이 High인 그룹 I 학습자 I, K, O, R, S 가



(그림 13) 스케줄링 결과

목요일 오후 3시에 협력학습을 하고, 지식 수준이 Mid인 그룹 II 학습자 A, C, D, E, F, G, J, L, Q가 목요일 오전 10시에 도형 학습을 할 수 있도록 스케줄링의 결과를 보여 주며 마지막 지식 수준이 Low인 그룹 III 학습자들 B, H, N, P, T, U, V는 목요일 12시에 학습을 할 수 있도록 스케줄링을 하고 있다. 이러한 스케줄링의 결과를 이용하여 각 동일 수준의 학습자들은 정해진 요일과 시간에 협력학습에 참여하게 된다.

CSP를 이용한 스케줄링의 결과는 일반 스케줄링 알고리즘으로 해결한 결과와 동일하지만 해를 구하는 속도에 있어 많은 차이를 가져왔다. 특히 학습자의 협력학습 참여시간이 변동이 되었을 경우 이를 재조정하는 시간이 일반적인 스케줄링 알고리즘을 사용했을 경우 보다 CSP를 이용하였을 때 매우 빠른 해를 보여주고 있다.

## 5. 관련 연구

본 연구와 관련된 연구는 분산 환경하의 협력 학습 시스템에 관한 연구와 동적인 스케줄링을 이용한 학습자의 학습 시간 스케줄링에 관한 내

용이 있다.

기존의 웹기반 협력 학습은 다양한 학습자들이 학습에 참여하여 학습할 수 있는 시스템의 구조와 다양한 멀티미디어 매체를 이용하여 동시에 학습을 하도록 지원하는 툴과 응용 프로그램에 관한 연구와 협력학습에 관계된 콘텐츠, 협력학습의 교수 방법에 대한 내용에 관한 것들이었다. 근본적인 학습자에 대한 이해에 관한 학습자 모델이나 학습자에 대한 진단 검사 방법에 관한 연구는 매우 부족하다.

기존의 ITS에 관한 연구를 살펴보면 ELM-ART, ASSERT, SHERLOCK 등이 있으며 학습자들의 스케줄링이나 진단 검사를 통하지 않고 학습을 원하는 모든 학습자들이 참가하여 지능적으로 학습할 수 있다. 이러한 시스템의 문제는 학습자 모델을 고려하고 있지 않고 또한 학습자들의 지식 수준을 고려하고 있지 않기 때문에 경직된 시스템이다.

기하학습에 관한 지능형 교수시스템의 경우 심임섭(1994), Kurt VanLehn(2000)의 Andes 시스템이 있으며 이 시스템은 학습 내용의 지식 표현 측면을 강조하였으며 단일 사용자를 위한 학습 시스템으로 제한되어 있다.

지능형 교수 학습 시스템의 경우 M.M. El-Khouly와 B.H.Far, Z.Koono(2000)등의 연구가 있으며 이 연구에서는 에이전트를 이용한 지능형 시스템의 구조에 국한하여 연구를 진행하였으며 이 연구는 학습자에 대한 연구를 제외하고 기초적인 시스템의 구조에 대해 논의되었다.

학습 시스템에 스케줄링을 적용한 경우는 기초적인 스케줄링 알고리즘을 이용한 경우가 대부분이며 CSP를 이용하여 스케줄링한 연구는 없다. 에이전트 기반의 CSP 적용 스케줄링을 이용하여 최단 시간 내에 최적의 해를 구할 수 있으며 기존의 방법으로 구한 해는 해결 시간이 너

무 오래 걸리기 때문에 쓸모 없는 결과가 되어 버리기 쉬운 단점을 포함하고 있다.

## 6. 결론

본 연구에서는 원격 협력 학습을 위한 ITS의 학습자 모델을 기초로 진단 에이전트와 도형 협력 학습을 가능하게 하는 스케줄링 에이전트를 설계, 구현하였다. 기존의 ITS가 지니고 있는 문제점들을 극복하고 웹 상에서 더욱 확장된 웹 기반 ITS의 구조를 설계하여 제시하였고, 에이전트를 기반으로 하는 새로운 형태의 지능형 교수 시스템의 세부 구조를 제안하였다. 진단 에이전트는 학습자의 도형에 관한 선행 지식 수준을 규칙 기반 시스템을 통해 진단했으며 추론 결과에 대한 신뢰도를 높이기 위해 시간과 가중치를 부여한 Temporal Weight 확신 인자 공식을 새롭게 제안하여 적용하였다. 에이전트를 기반으로 하는 스케줄링 에이전트를 설계 구현하여 수준별 협력 학습을 가능하게 하였으며, 또한 스케줄링 에이전트는 CSP기법을 적용하여 학습자의 요구사항과 제약 조건을 가지고 알맞은 시간을 할당하여 적절한 학습을 할 수 있도록 하였다. 또한 다중 학습자와 경쟁과 협력을 하며 최적의 학습자와 학습을 할 수 있도록 스케줄링을 하였다. 교수전략 모듈과 전문가 모듈은 분산화 되어 지식을 공유하며 확장 할 수 있는 구조를 가질 수 있으며 KQML을 이용한 서로간의 통신으로 다양한 학습자 모델을 새롭게 생성해 나갈 수 있다.

향후 연구 계획은 교수 전략 모듈과 전문가 모듈의 미흡한 부분을 개선 확장하고, 협력학습에 있어서 다양한 학습 방법에 대한 연구가 시급하다. 또한 협력 학습에서 나타나는 집단의

지식 수준 파악의 어려움과 집단 전체가 잘못 인식하고 있는 오류에 대한 진단, 교정 부분이 꼭 필요한 연구 분야이며 이를 위한 다중 분산 환경의 새로운 학습자 모델의 연구가 꼭 필요하다. 또한 다중의 학습자의 모델을 정립하여 개별학습과 지능적인 학습의 방법도 지속적으로 연구되어야 한다.

## 참고 문헌

- [1] Badrul H. Khan, "Web based Instruction", Educational Technology Publication, (1997),1-80.
- [2] B. A. Nadel, "Constraint Satisfaction Algorithms", Computational Intelligense, Vol.5, No 4, (1989), 188-224.
- [3] Baker, F.B "The Basic of Item Response Theory" Portsmouth, NH, Heinemann, 1985.
- [4] Barbara Hayes-Roth. "An Architecture for adaptive intelligent agent systems", Artificial Intelligent, Vol 72, (1995), 329-365.
- [5] Birnbaum, A, "Some latent trait models and their use in inferring an examine's ability", Addison Wesley, 1968.
- [6] David H. Jonassen, "Computer in the classroom : Mindtools for critical thinking", prentice hall, (1996), 47-95.
- [7] Derek H. Sleeman, "Inferring Student Models for Intelligent Computer Aided Instruction", Machine Learning, Morgan Kaufmann, (1983), 483-509.
- [8] E.H. Durfee and S. Sen, "A formal analysis of communication and commitment in distributed meeting scheduling", 11th Workshop on DAI, (1992), 333-344.
- [9] E. Tsang, "Foundations of Constraint Satisfaction", Academic Press, 1993.

- [10] Finin, T., Labrou, Y. and Mayfield, J., "KQML as an agent communication language", MIT Press, 1995.
- [11] Gerhard weiss, "Multiagent System", Mit press, (1999), 62-93.
- [12] Moises, L. and Thomas, D. "A Framework for the Development Architecture", IEEE Expert Dec, (1996), 47-59.
- [13] Marry A. Nixon, "Collaborative learning Web Based Instruction", AAAC, Nov., 1999.
- [14] M.M.El-Khouly, B.H.Far, Z.Koono, "Expert tutoring system for teaching computer programming languages", Expert systems with application 18, (2000), 27-32.
- [15] Nicholas R. Jennings, "Intelligent Agent", Springer Verlag, (1996),1-28.
- [16] Schulze, K.G., Shelby, R.N., Treacy, D.J., Wintersgill, M.C. (2000). Andes: A Coached Learning Environment for Classical Newtonian Physics. To appear in Proceedings of the 11th International Conference on College Teaching and Learning. Jacksonville, FL, April, (2000).
- [17] Shim Leem Seop, "Intelligent Tutoring System", Korea Information Science Society Review, (1994), 52-69.
- [18] Weber, G. & Specht, M, "ELM-ART:User modeling and adaptive navigation support in WWW-based tutoring systems". Proceedings of User Modeling '97 (1997), 289-300.