

전문가 시스템과 데이터 베이스를 이용한 이산 사건 시물레이션 모델링

A Discrete Event Simulation Modeling Using the Expert System and Database

김 형 중, 이주용, 조 대 호

(Hyung-Jong Kim, Ju-Yong Lee, Tae-Ho Cho)

Abstract : 시물레이션 모델과 전문가 시스템의 학제간 연구는 그 동안 많은 학자에 의해 진행되어 왔다. 전문가 시스템은 추론 기관과 지식베이스로 구성되며, 지식베이스는 사실과 규칙으로 구성된다. 사실과 규칙은 추론 기관의 추론을 위한 정보로 사용된다. 요즘의 정보 시스템은 데이터베이스를 가지며 데이터베이스의 정보를 기반으로 모든 처리가 진행된다. 이러한 정보 시스템에 삽입되어 사용되는 시물레이션 모델의 전문가 시스템도 데이터베이스의 정보를 사용하여 추론을 해야한다. 데이터베이스의 정보는 전문가 시스템의 추론을 위한 사실로 사용되고, 추론 결과를 저장하기 위한 저장 장소로 사용된다. 본 연구에서는 전문가 시스템의 사실과 데이터베이스 사이의 사상을 정리하고, 이를 위한 사실 클래스를 제안한다. 이 사실 클래스는 데이터베이스의 데이터를 전문가 시스템이 사용할 수 있도록 필터링하는 기능과 추론을 위한 의미 분석의 기능을 갖는다. 또한, 데이터베이스에 추론 결과를 저장할 때 생기는 데이터의 일치성과 무결성 문제를 해결한다. 이 사실 클래스를 갖는 전문가 시스템 기능을 갖는 시물레이션 모델을 그레이팅 생산 공정의 시물레이션에 적용하였다.

Keywords : Simulation Modeling, Expert System, Knowledge Base, Fact Class Library.
의 결합에서 생길 수 있는 문제점들을 해결하는

1. 서론

시물레이션 모델과 전문가 시스템의 복합 연구는 그 동안 많은 학자들에 의해서 진행되어 왔다[1]. 시물레이션 모델은 전문가 시스템과의 인터페이스를 통해서 지식 처리 기능을 소유할 수 있다. 또한, 전문가 시스템은 시물레이션 모델을 통해서 시간 기반의 정보를 추론할 수 있게된다[2][3][4].

최근 시물레이션 모델을 실제 시스템의 제어를 위해서 사용하게되는 추세에 따라서 시물레이션 모델에서 사용되는 전문가 시스템이 실제 시스템이 사용하는 데이터를 사용해야하는 필요성이 제기되고 있다. 대부분의 실제 시스템이 데이터베이스를 통해서 방대한 양의 정보를 관리하기 때문에 전문가 시스템이 이 데이터들을 사실로 사용하기 위해서는 이를 위한 인터페이스가 요구된다. 데이터베이스 시스템은 많은 양의 자료를 일치성과 무결성을 유지하면서 효율적으로 관리할 수 있고, 전문가 시스템은 추론기능과 유연성 있는 지식의 표현력을 갖는다. 이런 시스템들의 특징이 상호 보완을 이룰 때 지식처리의 향상을 도모할 수 있다[5][6].

본 논문에서는 전문가 시스템과 데이터베이스와

방법으로 FCL(Fact Class Library)을 제시하였다. 이를 통해 전문가 시스템과 데이터베이스 시스템의 인터페이스(Interface)를 위한 기반 기술을 제공하고자 한다. 또한, FCL을 사용하여 GPSS(Grating Process Scheduling System) 시물레이션 모델을 디자인 및 구현하여 FCL의 기능적 특성을 검증하였다.

2. 배경 이론

전문가 시스템은 적용 영역의 전문가들이 가지고 있는 전문 지식을 지식베이스로 구축하여 저장 관리함으로써 컴퓨터가 전문가의 기능을 인간과 같은 논리적인 사고로 대신 수행케 하는 시스템이다[5][7]. 전문가 시스템은 추론엔진과 지식베이스로 구성된다. 추론 엔진(Inference Engine)은 전문가 시스템의 전체적인 운영을 담당하는 기관으로 전문가들의 지식을 이용하는 방법을 정의하고 있다[5][7]. 지식베이스는 전문가의 지식을 담고 있는 지식의 저장 장소로 사실과 규칙으로 구성된다[8]. 사실은 지식 베이스의 지식을 적용하고

자 하는 현재의 상태이며 규칙은 문제를 풀어 나가기 위한 장기 정보(long-term information)로서 전문가의 전문적 지식을 통하여 새로운 사실이나 가정을 만들어 내는데 필요한 정보이다.

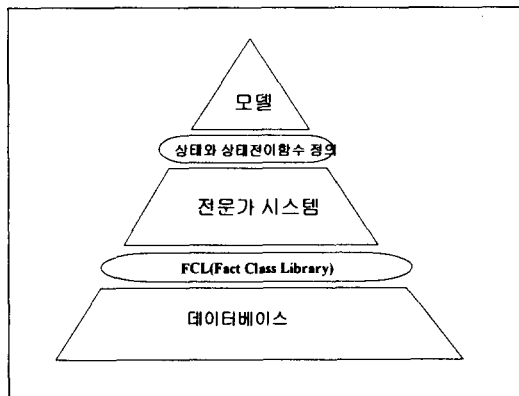
데이터베이스는 서로 연관이 있는 데이터들의 모임이며, 특정 의미를 가지는 데이터의 모임이다 [9]. 데이터베이스 관리의 주요 이슈(issue)로는 동시 사용에 의한 문제 해결, 잘못된 데이터의 방지, 데이터의 중복성 제거, 데이터의 보안 문제 등을 들 수 있다[5]. 특히 데이터베이스를 관리하는 관점에서 무결성 규정은 데이터베이스내의 데이터가 일관성의 유지를 위하여 충족시켜야 하는 제약 조건으로 데이터베이스를 정확하고 유효하게 유지하여 의미적 에러를 방지하기 위한 제약 조건이다.

3. 모델 설계 및 적용

본 논문에서는 FCL(Fact Class Library)을 제안하여 시물레이션 모델링에서 전문가 시스템과 데이터베이스의 인터페이스문제를 해결하고자 한다. 본 장에서는 FCL의 디자인 및 그레이팅 생산 공정의 시물레이션 모델링을 소개한다.

3.1 전문가 시물레이션 모델의 정보 처리 계층

<그림 1>는 전문가 시물레이션 모델의 정보 처리 계층도를 나타낸 그림이다. 전문가 시스템과 데이터베이스 사이, 시물레이션 모델과 전문가 시스템 사이에 각각 인터페이스를 위한 모듈이 존재한다. 전문가 시스템과 데이터베이스 간의 결합은 논문에서 제시하는 FCL이 인터페이스 역할을 하며, 시물레이션 모델과 전문가 시스템간의 연계는 모델 구축에서의 상태와 상태전이함수를 정의함으로 연계가 된다.

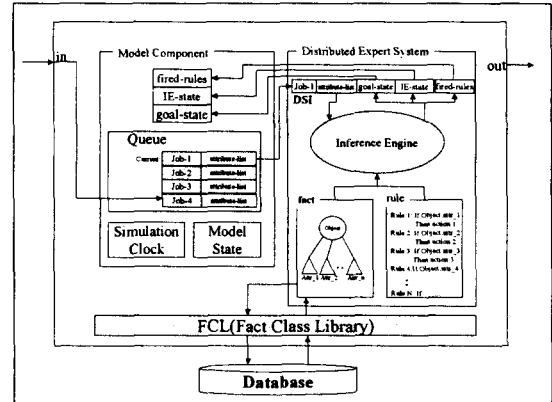


<그림 1> 정보 처리 계층도

3.2 FCL기반의 모델 아키텍처

<그림 2>는 FCL을 사용하는 시물레이션 모델의

구조이다. 본 논문에서 구현한 모델은 전문가 시스템이 시물레이션 모델에 삽입된 모형이고, 데이터베이스는 외부에 존재한다. 전문가 시스템에서 추론하기 위해 필요한 지식은 데이터베이스에서 FCL을 통해 얻는다. 그리고 전문가 시스템에서 추론하여 나온 결과와 삽입, 삭제 및 갱신된 데이터는 또한 FCL을 통해 데이터베이스에 반영된다.



<그림 2> FCL기반의 시물레이션 모델

3.3 인터페이스 설계

1) 시물레이션 모델과 전문가시스템

이산사건 시물레이션모델과 전문가 시스템간의 연계를 위해서는 모델에 추론 상태를 정의하고 상태 전이 함수를 정의함으로써 이루어진다. 시물레이션 모델에서의 상태는 정의된 특정 작업을 수행하게 하며, 시간에 따른 상태전이 함수에 의해서 상호작용(Interaction)하게 된다. 상태 전이 함수는 시간에 따른 상태 변화와 전이를 담당하며 상태와 상태간의 연결 통로로써 동작한다.

2) 전문가 시스템과 데이터베이스

전문가 시스템과 데이터베이스간의 결합에서 생길 수 있는 문제점으로 첫째, 데이터베이스내의 테이블의 각 필드(field)들이 전문가 시스템의 지식베이스의 사실(fact)에 반영될 때, 각 시스템의 지식 표현 방법이 다르기 때문에 지식의 타입(type)에 따른 클래스의 속성들과 올바른 사상이 되어야 한다. 둘째, 데이터베이스에서 추출된 데이터 중 추론에 필요한 데이터가 부재(Null)이거나, 그 데이터가 적합한지를 파악해야 한다.

전문가 시스템과 데이터베이스의 결합을 위해서는 우선, 전문가 시스템의 지식 표현 방법과 데이터베이스의 자료 표현 방법이 서로 다르기 때문에 시스템의 특성에 맞게 표현된 자료를 서로 사상하여 전환하는 작업이 이루어져야 한다. 두 구

조의 대응 관계는 <표 1>과 같다[7].

그리고 데이터베이스 시스템의 테이블을 구성하고 있는 행, 열, 셀들은 각각 지식베이스에 객체, 속성, 객체 슬롯(object slot)으로 사상되며, 역으로 지식베이스에 있는 객체, 속성, 객체 슬롯은 데이터베이스의 테이블에 행, 열, 셀(cell)로 사상시킨다. 왜냐하면, 인터페이스 대상이 되는 데이터베이스 시스템의 테이블은 지식베이스 내부의 클래스에 해당하기 때문이다.

<표 1> 데이터베이스와 사실의 사상

데이터베이스		사실 베이스
테이블(Table)	↔	사실클래스(Fact Class)
레코드(Record)	↔	사실(Fact)
필드(field)	↔	속성(Attribute)

3) FCL(Fact Class Library) 디자인

FCL은 데이터베이스에서 추론에 필요한 데이터를 추출하는 기능과 이 데이터를 전문가 시스템에서 추론하기 위한 사실(Fact)로 사상하기 위한 것이다. 이를 통해 데이터 레벨(level) 정보를 지식 레벨 정보로 한 단계 높이는 처리 과정이 수행된다.

FCL은 다음 4가지의 기능을 수행한다.

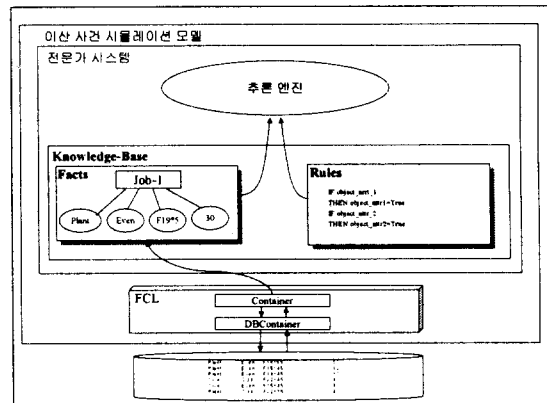
- 추론에 필요한 정보를 데이터베이스에서 추출: 추론을 위해 필요한 정보를 데이터베이스에서 가져오기 위한 작업을 한다.
- 추론에 필요한 데이터에 대한 의미부여: 데이터베이스에서 추출해온 정보를 추론을 위한 사실로 변환하는 작업을 한다. 데이터베이스의 제한된 데이터 타입으로 인해 표현할 수 없었던 정보를 추론을 위해 정의된 데이터의 형태로 변환한다.
- 추론에 필요한 정보를 얻기 위한 저장 공간: 추론에 사용되는 정보를 얻고, 추론 결과를 저장하기 위한 저장 공간이 된다.
- 추론 결과 저장 시 데이터베이스 객체의 무결성, 일치성, 식별성의 유지: 데이터베이스에 추론 결과를 저장할 때 무결성, 일치성, 식별성을 유지시켜주는 기능을 한다.

FCL은 <그림 3>와 같이 Container Class와 DBContainer Class로 구성된다.

첫째, DBContainer는 데이터베이스 입출력에 관련된 모든 프리미티브(primitive) 함수들을 정의해 놓았고 질의(query)에 의해 필터링된 데이터를 가져오는 기능을 갖는다. 또한, 추론 결과를 저장할

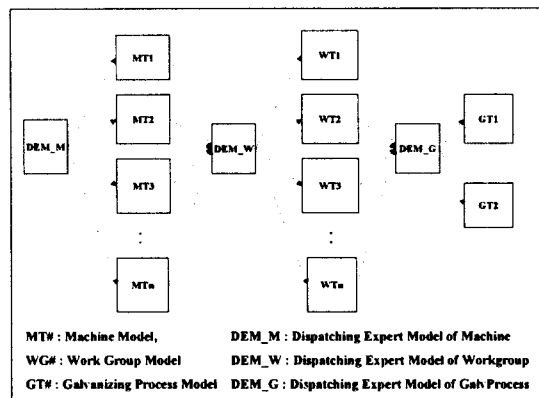
경우 데이터베이스 객체의 제약조건을 맞추어주는 기능을 갖는다. 둘째, Container에는 데이터베이스를 전문가 시스템이 추론에 필요한 지식을 얻는 지역 메모리처럼 사용하기 위한 저장 버퍼로서 데이터의 의미를 해석하여, 전문가 시스템의 사실로 사용될 수 있도록 변환한다.

예를 들어, 특정 머신의 정보로 추론을 해야 할 경우 DBContainer는 주어진 조건에 맞게 데이터베이스에게 질의를 하여 가져오는 기능을 수행하고, 가져온 데이터 중에 베어링바(bearingbar)의 타입이 'f19*3'이라는 데이터는 각 토큰 별로 'f'는 flatbar, 높이는 '19' 그리고 두께는 '3'의 형태로 변환하는 일을 Container가 한다. 이렇게 변환된 사실들은 Container에 멤버 리스트의 형태로 저장되어 추론에 사용된다.



<그림 3> 전체 추론 과정

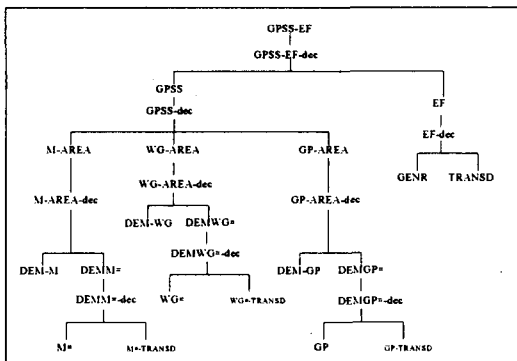
3.4 GPSS 모델에의 적용



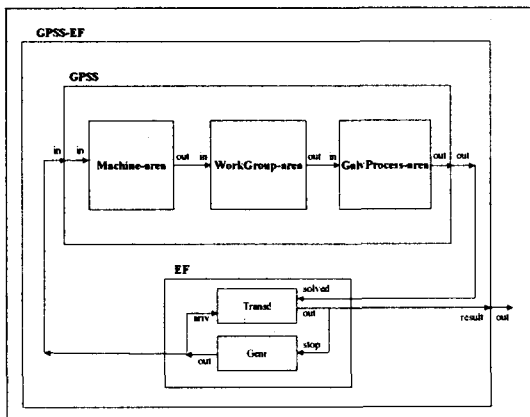
<그림 4> Overall Structure of GPSS Model

<그림 4>은 본 연구에서 구현한 시물레이션 모델의 전체적인 구조이다. 모델을 보면 DEM(Dispatching Expert Model), M#(그래이팅을 만들기 위한 머신 모델), WG#(엔드 바 용접을 위한 가공 팀 모델), GP#(도금 공정)으로 구성된 것을 알 수 있다. <그림 4> 모델은 목적에 따라

분류하면 프로세스 모델을 구성하는 M#, WG#, 그리고 GP#과 프로세스 모델의 지능형 제어를 위해 존재하는 DEM모델로 구분된다. DEM-M 모델은 여기서 받은 각 머신의 현재 상황과 작업의 내용을 보고 최적의 머신에 작업을 할당한다[14]. DEM은 모델 안에는 작업을 프로세스 모델에 할당하는데 필요한 규칙들이 rule-base에 저장되어 있다. DEM 모델은 rule-base의 규칙과 작업 정보에 기초해서 최적의 모델에게 작업을 할당한다. 그레이팅 프로세스 스케줄링을 위한 모델들과 experimental frame의 구조는 <그림 5>에 나타나 있다. Root-entity인 GPSS-EF는 Experimental Frame(EF)와 GPSS-DES로 구성되어 있다. GPSS-DES는 <그림 5>의 모델의 구조를 의미한다. GPSS-DES entity는 M-AREA, WG-AREA, 그리고 GP-AREA 3의 구성 요소로 이루어지고, 각각의 구성 요소들은 하나의 DEM(Dispatching Expert Model)과 각 공정의 프로세스 모델로 구성된다. 특히, 각각의 프로세스 모델에는 성능을 점검하기 위한 M#-Transd, WG#-Transd, GP#-Transd 모델을 연결하였다.



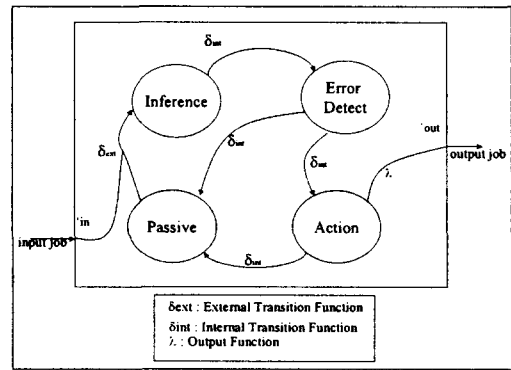
<그림 5> System Entity Structure(SES)



<그림 6> GPSS-EF의 구조

<그림 6>은 GPSS-EF 모델에서 GPSS 모델과

EF 모델과의 컴포넌트간의 연결 관계를 볼 수 있다. GPSS-EF 모델은 GPSS모델과 EF 모델로 구성되어 있으며 GPSS 모델은 Machine-area 모델, WorkGroup-area 모델, GalvProcess-area 모델로 구성되어 있다. EF 모델에서는 Transd와 Genr로 구성된다.



<그림 7> State Transition Diagram

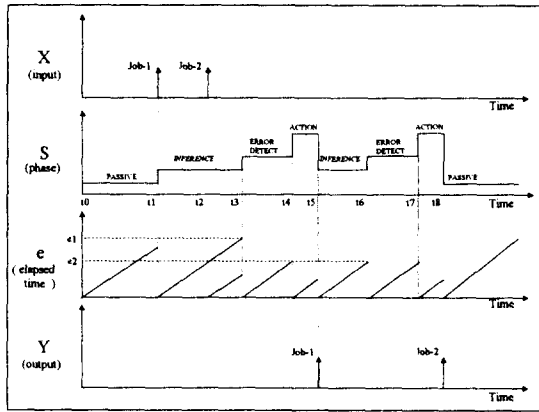
<그림 7>은 GPSS시스템에서의 상태 전이 다이어그램을 보여주고 있다. Passive, Inference, Error Detect, Action의 4개의 state를 가지고 있다. Passive상태는 현재 모델에 처리되고 있는 작업이 없고, 작업을 처리할 준비가 되어 있음을 나타낸다. Inference 상태는 모델 안의 전문가 시스템이 현재 작업을 처리하기 위한 추론을 하고 있음을 나타낸다. 이 상태에서는 모델 안의 전문가 시스템이 제어를 가지고 있다. Passive에서 Inference로의 상태 전이는 입력포트인 'in'을 통한 외부 입력에 의해서 구동되는 외부 전이 함수에 의해서 이루어진다.

정상적인 추론이 끝나면 모델의 상태는 Inference에서 Error Detect로 전이된다. Error Detect상태에서는 추론 결과를 데이터베이스에 저장하기 전에 FCL에 있는 데이터의 무결성을 검사한다. 무결성 검사는 주키(primary key)가 널(Null)이거나 동일 주키에 2개 이상의 결과 값이 있는지를 검사하게 된다. 정상적으로 추론되었다면 Error Detect상태로 전이하고, 만일 비정상적으로 추론되었다면 에러가 발생하였다는 정보를 log에 기록하고 Passive상태로 전이한다.

Action상태에서 모델은 추론의 결과에 따라 동작한다. Action상태는 모델링 하는 대상에 따라 다른 이름의 여러 상태로 나누어질 수 있다. Action상태가 종료되면 DEVS-Component의 output함수가 구동되어 출력을 제공한다.

<그림 8>는 timing diagram의 한 예를 보여주

고 있다. 입력을 보면 한 가지 타입으로 처리되어야 할 작업을 나타낸 것을 볼 수 있다. 첫째는, 처리되어야 할 작업이고 t1이라는 시간에 외부 입력 job-1이 Passive상태에서 모델 안으로 입력 되면 모델의 phase는 inference상태로 전이된다.



<그림 8> Timing Diagram of GPSS

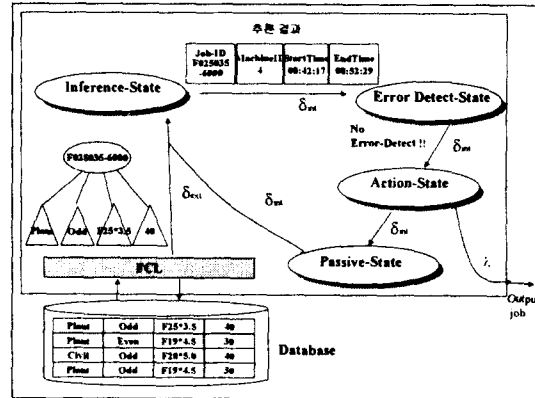
그리고, t2에서는 job-2가 입력으로 들어오고, 이 job은 현재 모델이 처리할 수 없는 상태이기 때문에 모델의 큐에 삽입된다. 전문가 시스템의 추론이 정상적으로 종료되어, 이 추론의 결과에 기초해 작업을 처리한다. 이후 t3에서 Error Detect상태로 전이한 후 추론 결과에 대한 에러를 검사한다. t4에서 Action상태로 전이한 후 t5에 종료되고 동시에 추론의 결과로 job-1이 출력으로 나간다. 그리고, 모델은 현재의 큐의 상태를 점검하여 t2에 입력된 job-2를 처리하기 위한 inference상태로 전이된다. 시간 t6에 추론이 정상적으로 종료되고Error Detect상태로 전이한 후 추론 결과에 대한 에러를 검사한다. 모델의 상태는 다시 Action상태로 전이되어 Action상태가 종료되는 t8에는 job-2를 출력한다. <그림 8>을 볼 때 job-1과 job-2를 처리하기 위해서 사용된 inference 시간은 e1과 e2로 나타낼 수 있다. 이 추론 시간은 전문가 시스템의 규칙을 실행(firing) 하는데 걸리는 시간이라고 할 수 있다.

4. 시물레이션 모델의 기능 검증

이 장에서는 4장에서 디자인 및 구현된 시물레이션 모델이 정상적인 추론을 했을 때와 비정상적인 추론을 했을 때의 모델의 검증과정을 설명한다. 아래 <그림 9>은 시물레이션 모델의 정상적인 추론 과정을 검증하는 그림이다.

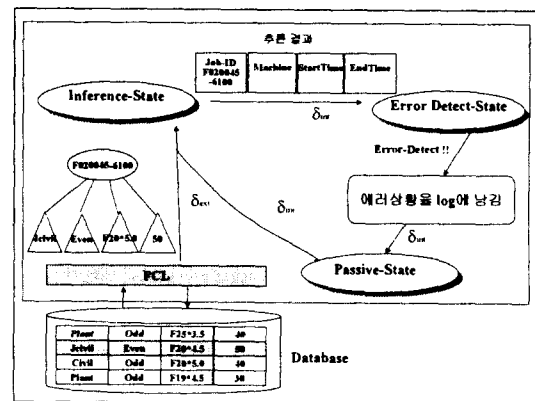
<그림 9>에서 보는 바와 같이, 모델이 추론을 하려면 데이터베이스에서 추론에 필요한 데이터를

추출한다. 추출된 데이터가 FCL에서 DBContainer와 Container를 거치면서 데이터가 지식베이스의 사실로 객체지향적으로 사상되어 사실클래스를 이루는 객체가 된다. 이러한 객체는 Inference상태에서 추론되고 추론된 결과는 Error Detect상태에서 추론이 정상적으로 되었는지 검사를 하게 된다. 정상적으로 추론되었다면 Action 상태로 전이하여 결과 값을 출력하게 된다.



<그림 9> 정상 상태 추론

<그림 9>에서처럼 추론의 결과로 4번 Machine으로 추론 결과가 나오게 된다. 이러한 과정은 데이터베이스에서 추출된 데이터가 없을 때까지 진행된다. ErrorDetect 상태에서는 이러한 추론 결과에 대한 검사를 하게 된다. 이때 행하는 검사는 정상적인 추론이 되었는가와 무결성 검사이다. ErrorDetect 상태에서 에러가 검출되지 않으면 Action상태로 전이되며 추론의 결과를 출력하게 된다.



<그림 10> 비정상 상태 추론

만일 <그림 10>에서처럼 추론한 데이터가 없거나 일관성이 지켜지지 않아 Error Detect상태에서 비정상적인 추론이라고 감지되면 Log에 에러가 발생하였음을 남기고 Passive상태로 전이하게 된다.

다. GPSS에서는 Machine의 용도가 'Plant'이거나 'Civil'인 것만 처리하므로 <그림 10>에서 보는 바와 같이 Machine의 용도가 'Jcivil'라면 Machine에 대한 추론이 불가능하므로 추론 결과는 없다. 이렇게 비정상적인 추론 결과를 Error Detect상태에서 에러임을 감지하면, 에러처리를 해당 job의 에러를 log에 쓰고 Passive상태로 전이한다. 이렇게 모델의 검증을 정상추론과 비정상 추론에 대한 각각의 예를 통해 검증하였다.

5. 결론 및 향후과제

본 연구에서 GPSS를 모델로 전문가 시스템과 데이터베이스를 이용한 시뮬레이션 모델을 구축하고, 지식 처리 모듈이 시뮬레이션 모델에 제대로 표현되는지에 관한 타당성을 검증하기 위한 방법을 검토하였다. 모델을 구축하면서 전문가 시스템과 데이터베이스의 결합에서 생길 수 있는 문제를 해결하기 위한 FCL를 디자인하고 FCL을 사용하여 모델링하였다. 본 논문에서 제시한 FCL은 기타 다른 데이터 소스와 전문가 시스템과의 인터페이스에도 발전시켜 적용 가능 할 것으로 본다.

향후 연구 과제로는 본 논문에서 모델에 대한 검증만을 하였지만 장치 성능 면에서 모델의 분석이 이뤄져야 하고, 'Error Detect'상태에서 Error의 교정 기능 등이 구현되어야 하며, 시간 기반이 변수들의 유효 여부에 기초해 정확한 추론을 하는 것과 시간 의존 변수의 유효성을 점검하기 위한 제어 모델을 사용하여 정보의 유용성 여부를 시뮬레이션 결과에 반영시키는 연구가 필요하다. 그리고 FCL의 기능중 하나인 에러 검출에 대한 처리를 발전시켜 실제 시스템에 적용되는 설계 방법론이 정립되어야 할 것이다.

참고 문헌

- [1] Arons, H. De Swann, "Expert Systems in the Simulation Domain," Mathematics and Computers In Simulation, Vol, XXV, 1983.
- [2] 조대호, 김형종 "분산전문가 시스템의 기능을 갖는 이산사건 시뮬레이션" 한국시뮬레이션학회 논문지, Vol. 7, No. 2, December 1998.
- [3] Bernard P. Zeigler, Tae H. Cho, Jerzy W. Rozenblit, "A Knowledge-Based Simulation Environment for hierarchical Flexible Manufacturing" IEEE Transaction On Systems, Man, and Cybernetics-Part A : System and humans, Vol. 26, No.1 January 1996
- [4] Tae H. Cho and Bernard P. Zeigler, "Simulation of Intelligent Hierarchical Flexible Manufacturing : Batch Job Routing in Operation Overlapping" IEEE Transaction On Systems, Man, and Cybernetics-Part A : System and humans, Vol. 27, No.1 January 1997.
- [5] Stonebraker, M., "Implementation of Rules in Relational Database System," Database Engineering, Vol.6, No.4, 1983.
- [6] 김형종, 조대호, 이철기, 김훈모, 노용환 "반도체 생산라인에서의 이탈처리 추적 전문가 시스템의 지식베이스 구축," 제어 자동화 시스템공학 논문지 제5권 제1호 1999.1.
- [7] R. Rodriguez, et. al, "Efficient Expert System: Rule-Base management Via Relational Database Techniques," Advances in Artificial Intlligence Research, Vol.1 pp. 65-79, 1989.
- [8] Kellogg,C., "From Data Management to Knowledge Management," IEEE Computer, Jan.,1986.
- [9] Stonebraker, M., Woodfill, J., and Andersen, E., "Implementation of Rule in Relational Data Base Systems," Database Engineering, Vol. 6, No. 4, 1983, pp.65-74.
- [10] Cremers, A. and Domann, G., "AIM-An Integrity Monitor for the Database System INGRES," Proc. Int. Conf. on Very Large Databases, 1983, pp.167-170.
- [11] Dittrich, K.R. Kotz, A.M., and Mülle, J.A., "An Event/Trigger Mechanism to Enforce Complex Consistency Constraints in Design Databases," ACM SIGMOD RECORD, Vol. 15 No. 3, Sept. 1986, pp. 22-36.
- [12] E. N. Gehani, et. al, "Event Specification in an Object-Oriented Database," Proceedings of ACM-SIGMOD, pp. 81-90, 1992.