객체지향방식에 의한 자동화제조시스템 시뮬레이터의 설계 및 구현*
Development of a Simulator for Automated Manufacturing Systems

이진규**, 이진환**, 이태억**, 오부경***, 오석찬***
Jin-Kyu Lee**, Jin-Hwan Lee**, Tae-Eog Lee**, Poo-Keong Oh***, and Seok-Chan Oh***

한국과학기술원 산업공학과**, 대우정보시스템***
Department of Industrial Engineering, KAIST**
Daewoo Information Systems Co. Ltd.***

Abstract

We discuss development of a simulator for automated manufacturing systems (AMSs) which have sophisticated automated material handling equipments and complicated work flows. The simulator is designed to satisfy the following requirements. A user should be able to easily configure or specify an AMS through a graphical user interface (GUI) and minimal data input. The user should be able to model diverse and complicated control logic for automated material handling systems like automated guided vehicle (AGV) systems, robot workcell systems and conveyor systems as well as complicated job flow control. Further, the user model should be easily verified and mapped into an executable simulation program. Real time animation is desired. Finally, the simulator should be easily maintained and extended. To satisfy the requirements, we use an object-oriented paradigm for modeling, designing, and programming of the simulator. We use an object-oriented modeling framework to design the modeling elements library, and take the process interaction approach for scheduling processes and events. To model a user-defined diverse control logic, we also develop a script language and its interpreter. We explain design and implementation strategies. We implement the simulator using Visual C++ 4.2 and Open GL on Windows NT and the Windows95. Some modeling examples will be demonstrated.

주요어: 자동화제조시스템, 시뮬레이터, 객체지향방식, 스크립트언어, 모델링요소라이브러리, SOOSM (Server and Object Oriented Modeling Framework)

1. 서 론

컴퓨터 시뮬레이션은 제조시스템의 설계단계시 효율적인 장비의 구성과 배치(layout) 및 용량(capacity)에 대한 의사결정 뿐 아니라, 운영단계시 제반의사결정에 필요한 사전 평가도구로써 널리 사용되고 있다. 특히, 물류시스템간의 복잡한 상호작용을 특징으로 하는 자동화제조시스템의 다양한 운영규칙을 비교평가하는데 있어서는 컴퓨터시뮬레이션이 최적의 도구로 평가받고 있다. 이러한 유용성에도 불구하고 시뮬레이션은 제조시스템 현장에 있는 FA(Factory Automation) 엔지니어에 의하여 널리 사용되지 못하고, 오직 소수의 전문가에 의해서만 유용하게 사용되고 있다. 그 이유는 대략 다음과 같이 세 가지로 요약할 수 있다. 첫째, 당면한 문제상황의 모델링을 위해서는 모델링에 대한 상당한 경험과 지식이 필요하다는 점이다. 일부 시뮬레이터 (혹은 시뮬레이션 언어)는 해당 시뮬레이터에 적합한 고유의 모델링 도구를 제공하고 있기도 하지만, 일반적으로 FA 엔지니어가 특정 모델링 방법론에 익숙해지기 위해서는 상당량의 학습을 필요로 한다. 둘째, 모델링 후 이를 시뮬레이션 프로그램으로 구현하기 위해서는 시뮬레이션 언어에 대한 심도 깊은 지식이 필요하다. 특히, 현재 구현한 시뮬레이션 프로그램의 유효성(validity)을 검증(verify)하고 수정하여 문제상황에 적합한 최종모델을 얻기 위해서는 시뮬레이션 언어의 사용 경험이 풍부해야 한다. 시뮬레이션 프로그램의 유효성 검증과정은 시뮬레이션기법을 실제 문제에 적용하는데 있어 가장 중요한 단계중 하나이다. 일반적으로 최초의 해당 시뮬레이션 프로그램을 구현하기 위하여 소요된 것 보다 더 많은 양의 시간과 노력이 이 단계에서 소요된다. 셋째, 시뮬레이션

*이 연구는 대우정보시스템과 통산산업부의 지원에 의하여 수행되었음.

에 필요한 입력데이타를 수집하여 이에 대한 근사 입력분포의 추정뿐 아니라, 반복하여 수행된 시뮬레이션 실행결과를 분석하고 해석하여 적절한 결론을 얻기 위해서는 상당한 수준의 확률과 통계에 대한 지식이 필요 하다. 일반적으로 이 점은 FA 엔지니어들이 간과하기 쉬운 사실이다. 입력데이타에 대한 근사분포가 부적합 경우의 시뮬레이션 결과는 아무런 의미가 없다(Garbage input, garbage output). 또한 유의미한 결과를 얻기위한 시뮬레이션 반복횟수의 결정이나, 실행결과에 대한 적절한 해석을 위해서는 통계적 지식이 절대적으로 필요하다.

본 연구의 목적은 시뮬레이션에 대한 경험과 지식이 거의 없는 FA 엔지니어들이 자동화제조시스템의 성능평가에 시뮬레이션기법을 쉽게 적용할 수 있도록 하기 위하여 사용이 편리한 제조시스템용 시뮬레이터(이하 K-SIM)를 개발하는 것이다. 자동화제조시스템용 시뮬레이터의 개발은 자동화제조시스템을 구성하는 일반적인(generic) 요소를 모델링라이브러리화 하기 위한 모델링기술, 절차적인 프로그램언어를 사용하여 병존하는 사건을 모사하는 기술, 다양한 제조시스템의 운영규칙을 모델링하는데 필요한 스크립트 언어의 설계기술, 빠른 속도의 실시간 애니매이션기술, 효과적이고 신뢰성 있는 확률분포의 생성 및 추정기술과 GUI 기술 및 각 구성모듈간의 통합화를 위한 소프트웨어 설계기술 등 다양한 분야의 관련 지식과 기술을 필요로 한다. 따라서 제조시스템용 시뮬레이터 개발에는 여러 분야에 대한 연구 주제가 있을 수 있으나, 본 연구에서는 객체지향적 시뮬레이터의 구조 및 각 모듈의 기능, 그리고 시뮬레이션 모델링요소라이브러리 구축을 위한 모델링과 구현방법에 한정하여 설명한다.

2. K-SIM의 개발전략

2.1 시뮬레이터의 개발동향

최근의 제조시스템용 시뮬레이터의 개발방향은 사용자가 시뮬레이션 모델을 쉽게 개발할 수 있도록 하기 위한 편리한 사용자인터페이스에 대한 강조와 시뮬레이션 모델의 유효성을 쉽게 검증할 수 있도록 지원하기 위한 실시간 애니매이션 기능의 강화, 그리고 기능확장성 (예를 들면, 시뮬레이션 모듈을 기반으로 하는 스케쥴링 시스템) 등으로 요약될 수 있다. 편리한 사용자인터페이스는 시뮬레이션 프로그램 개발기간의 단축과 밀접한 관계가 있다. 넓은 의미의 편리한 사용자인터페이스에는 자동화제조시스템의 복잡한 운영규칙의 손쉬운 모델링을 지원하기 위한 스크립트언어[1, 5]의 구비와 시뮬레이션 모델의 입력데이타에 대한 분포추정 기능 및 실행결과에 대한 통계적 해석지원 기능 등도 포함된다. 실시간 애니매이션 기능은 구현된 시뮬레이션 모델의 오류와 적합성을 검증하기 위한 가장 효과적인 방법이다. 만약 실시간 애니매이션 기능이 지원되지 않는다면, 사용자는 일단 구현된 시뮬레이션 모델을 장시간 실행시킨 다음, 그 실행결과로서 수집된 통계량을 해석하여 모델의 오류 및 적합성을 검증해야 한다. 이러한 경우에는 시뮬레이션 수행을 위하여 많은 시간이 소요될 뿐 아니라, 실행결과의 해석을 위하여 상당한 수준의 경험과 통계에 대한 지식이 필요하다. 이에 반하여 실시간 애니매이션에 의한 모델의 적합성 검증은 시뮬레이션 실행과 동시에 시뮬레이션 모델의 상태변화를 눈으로 확인할 수 있으므로 시뮬레이션에 대한 경험이 없는 사용자도 쉽게 모델의 오류를 발견할 수 있다. 결과적으로 실시간 애니매이션 기능은 시뮬레이션 프로그램 개발기간을 단축시키는 중요한 역할을 수행한다.

최근의 제조시스템용 시뮬레이터의 개발동향은 크게 두 가지로 구분할 수 있다. 하나는 SIMAN/CINEMA, SIMAN/ARENA, SLAM/SYSTEM 등과 같이 기존의 시뮬레이션 언어를 기반으로 하여 자동화공장의 시뮬레이션 모델을 쉽게 구현할 수 있도록 사용자인터페이스 기능을 강화하고 애니매이션 기능을 추가한 제품들이다. 이들은 SIMAN 이나 SLAM 등과 같이 이미 검증된 언어를 사용함으로써 시뮬레이터의 안정성을 확보하고 또한 기존의 SIMAN 이나 SLAM 을 이용하여 개발된 시뮬레이션 모델을 재사용할 수 있다는 장점이 있다. 이와는 달리 개발 초기부터 아이콘 바탕의 모델링 방법을 채택하여 사용자의 프로그래밍 노력을 최소화시킨 FACTOR/AIM, PROMODEL, AUTOMOD, Taylor II, SIMPLE 등이 있다. 이와 같은 GUI 및 데이타 입력에 의한 시뮬레이션 모델 개발방식은 시뮬레이션 프로그램의 개발 노력을 극소화하여 시뮬레이션 기법을 제조시스템 분야에 보급하는데 크게 기여하였다. 이와 같은 유형의 시뮬레이터는 모델 구성의 유연성을 보완하기 위하여 C 와 같은 범용프로그래밍 언어와의 링크기능이나 또는 자체의 스크립트언어를 제공하기도 한다.

한편 범용시뮬레이터의 최근 개발동향은 통칭 “객체지향적 시뮬레이션” 기법의 활용에 초점이 모아지고 있다. 이는 최근의 객체지향형 프로그래밍언어와 모델링 기법의 보급에

따른 결과이다. 객체지향형 모델링 및 프로그래밍을 지원하는 시뮬레이션 언어로는 SMALLTALK, MODULA II, MODSIM 등이 있으며, SIMPACK, DISC++, Sim++ 등과 같이 C++를 사용하여 시뮬레이션에 편리하게 사용할 수 있도록 라이브러리를 구현한 제품도 있다. 그러나 이러한 라이브러리들은 시뮬레이션에 대한 상당한 조예가 있는 사람만이 사용할 수 있다는 단점이 있다.

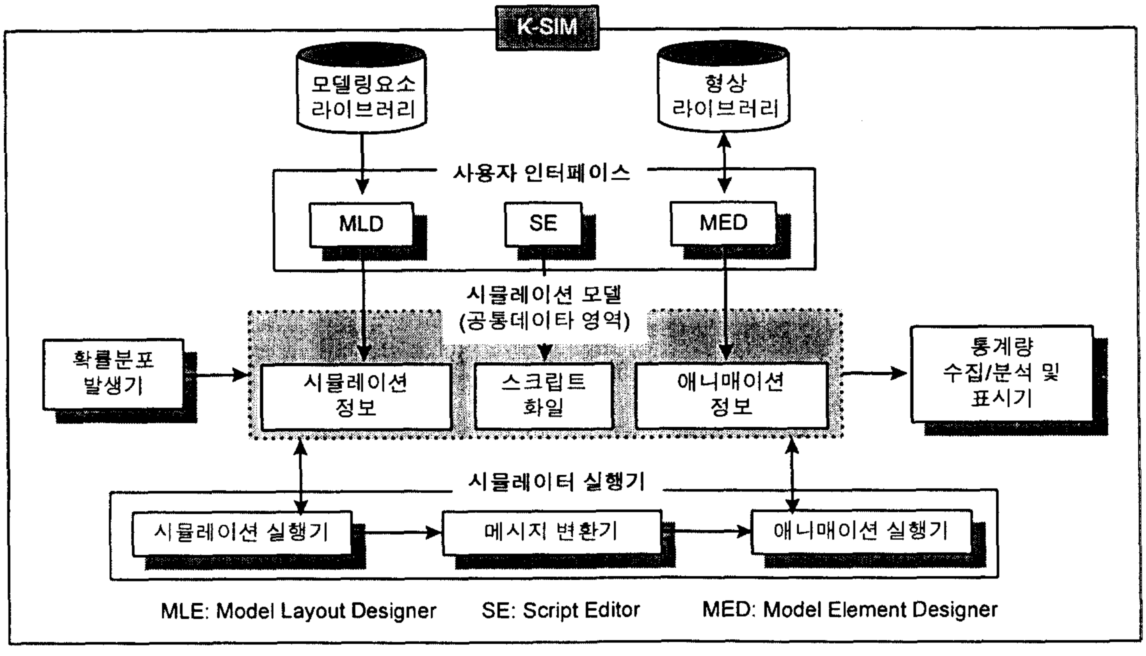
2.2 K-SIM의 개발전략

K-SIM의 개발에 있어 특히 중요하게 고려한 특성은 사용의 편리성과 모델링의 유연성, 그리고 높은 수준의 애니메이션으로 요약된다. 각각의 중요성에 대해서는 이미 앞절에서 언급하였다. 사용의 편리성은 두 가지 측면에서 고려하였다. 하나는 모델의 구축단계에서의 편리성이고, 다른 하나는 시뮬레이션 분석단계에서의 결과 분석의 용이성이다. 모델 구축의 편리성을 위하여 K-SIM은 아이콘과 대화상자를 이용한 모델링을 기본으로 한다. 즉 사용자는 K-SIM에서 제공하는 모델링요소라이브러리에서 필요로하는 구성요소를 선택하여 이들의 속성 값을 시스템 상황에 맞게 정의하는 것으로 대상 시스템을 구성하는 물리적 요소를 쉽게 모델링 할 수 있다. K-SIM에서는 3차원 실시간 애니메이션을 지원한다. 현재는 3차원 실시간 애니메이션이 시뮬레이터의 실행속도를 저하시키는 주요원인이 되고 있으나, 향후 애니메이션 기능을 자동화제조시스템의 각 요소간의 간섭확인 등에 사용할 수 있도록 확장성을 염두에 두고 애니매이션 모듈을 설계하였다. 또한, 자동화제조시스템의 다양한 운영규칙을 모델링하기 위하여 자체의 스크립트언어도 제공한다. 스크립트언어 및 이의 해석기 역시 K-SIM을 가상공장 수준으로 확장하기 위하여 구비해야 할 필수요소이다.

개발단계의 생산성과 유지보수의 용이성 및 향후 확장성 등을 고려하여 K-SIM 개발에는 객체지향 패러다임을 채택하였다. 즉, 시뮬레이터의 설계에 있어 각 모듈간의 독립성을 강조하였고, 특히 자동화제조시스템의 각 구성요소에 대한 모델링 라이브러리리를 구축하는데 있어서는 객체지향적 모델링기법(Object-oriented Modeling Technique)을 사용하였으며, 클래스(class) 계층간의 속성상속(inheritance) 및 정보은닉 등의 객체지향 프로그래밍 기법을 유용하게 활용하였다. K-SIM은 기본적으로 Windows NT 상에서 Visual C++ 4.2를 사용하여 개발하였으며 형상모델과 애니매이션 모듈은 Windows NT에 기본적으로 제공하는 Open GL을 사용하여 개발하였다. 또한 스크립트언어의 개발을 위하여 Lex를 이용하여 어휘분석기를 생성하였고, Yacc를 구문분석생성기로 사용하였다[1].

2.3 K-SIM의 구조

K-SIM의 주요 구성모듈은 다음 <그림 1>과 같으며, 시뮬레이션이 진행되는 동안 지속



<그림 1> K-SIM 구조의 개념도

적으로 데이타 변화을 관리하는 모듈로 구성된 주요부와 시뮬레이션모델 구현단계나 시뮬레이션 실행이 종료된 후 통계량 관리 등에 사용되는 사용자지원부, 모델링요소라이브러리 및 스크립트언어 지원 모듈 등의 부속모듈 등으로 편의상 구분할 수 있다. 시뮬레이터의 주요부는 전체 시뮬레이터의 제어에 필요한 핵심요소로서, 시뮬레이터의 각 모듈간의 제어를 담당하는 주제어모듈(main controller module)과 시뮬레이션 수행을 담당하는 시뮬레이션모듈(simulation module), 3차원 그래픽과 실시간 애니메이션을 담당하는 애니메이션모듈(animation module), 그리고 시뮬레이션모듈과 애니메이션모듈 간의 데이타 변환을 담당하는 변환모듈(translation module) 등으로 구성되어 있다. 사용자지원부는 대상시스템을 구성하는 물리적 구성요소의 배치 및 필요한 데이타를 입력하는 기본적인 환경을 제공하는 MLD (Model Layout Designer), 물리적 구성요소의 3차원 형상을 편집하는데 사용하는 MED (Model Element Designer), 시뮬레이션 실행과 함께 중요한 통계량의 변화추이를 표시하는 실시간 통계표시기, 그리고 수집된 통계량의 분석에 사용하는 통계분석기(SAFE, Statistical Analyzer for Factory Evaluation)와 제반입출력 환경을 제공하는 GUI 모듈로 구성되어 있다. K-SIM을 구성하는 주요 부속모듈에는 시뮬레이터의 실행 환경이나 조건의 정의 및 복잡한 운영규칙의 모델링에 필요한 스크립트해석기와 제조시스템을 구성하는 주요 요소의 속성(attribute)과 동적특성(behavior)을 정의하여 라이브러리로 구성한 모델링요소라이브러리, 그리고 모델링 요소의 형상을 저장한 형상라이브러리 등이 있다.

3. 객체지향형 시뮬레이션프로그램 구현 방법론

3.1 SOOSM의 개요

K-SIM은 자동화제조시스템의 모델링 방법론으로 SOOSM(Server and Object Oriented Simulation Modeling Framework)을 사용하였다[3]. 이는 제조시스템의 구성요소를 가공이나 조립 등의 서비스를 제공하는 서버(가공장비, 버퍼, AGV 등)와 서비스를 제공받는 작업물, 서버가 서비스 수행을 위해 필요로 하는 기타 자원(resource, 팔렛이나 기타 조립에 필요한 요소들), 작업물의 가공순서와 필요한 자원에 대한 정보를 저장하는 정보객체(information object, 프로세스플랜, 공구 데이타베이스 등), 그리고 AGV나 AS/RS 등의 제어에 필요한 제어객체 등으로 파악한 후, 서버를 중심으로 프로세스간 상호작용방식[2, 4]의 시뮬레이션모델을 구현하기 위한 방법이다.

제조시스템의 구성요소를 라이브러리화하기 위해서는 우선 각 구성요소의 속성과 동적특성을 정의한 후, 이를 바탕으로 유사한 속성이나 동적특성을 갖는 모델링 요소들 사이의 속성상속 관계를 나타내는 클래스계층도(class hierarchy)를 작성한다[8]. 이때 향후 프로그램의 유지보수의 편의를 위하여 가상클래스를 정의하면 효과적으로 클래스계층도를 관리할 수 있다. 제조시스템의 동적특성을 모델링하는데 있어 정식표현방법(formal specification)에 익숙하지 않은 개발초기에는 Event Schema와 같은 비정식표현방법(informal specification)을 사용할 수도 있으나, 가급적 정식표현방법을 사용하여 모델링하는 것이 유리하다[3]. 특히 프로세스간 상호작용에 대한 모델링은 일종의 프로토콜에 해당하므로 Automata나 Petri net과 같은 정식표현방법을 사용하면 모델의 오류검증을 위해 재사용할 수 있는 장점이 있다[6].

3.2 객체지향적 시뮬레이션엔진의 구현

앞에서 언급한 바와 같이 K-SIM의 시뮬레이션 모듈은 서버중심의 프로세스간 상호작용방식으로 개발하였다[2, 4]. 프로세스간 상호작용이라 함은 서버간에 필요한 정보 교류나 작업물 전달 등을 가능하면 제조시스템 현장에서 사용하는 방식에 가깝게 구현하는 것을 의미한다. 예를 들면, 이웃한 서버간의 작업물을 전달하는 데 있어서 공통의 게시판(blackboard)을 이용하지 않고, 실제 신호를 주고 받는 방식에 의하여 전달하는 것을 의미한다. 물론 중앙제어부(main controller) 등과 같은 제어객체에서는 게시판 방식을 사용한다. 이러한 프로세스간 상호작용 방식은 향후 K-SIM을 공장용 통신프로토콜 개발에 활용할 수 있는 가능성을 내포한다.

프로세스간 상호작용방식의 시뮬레이션 모듈을 개발하기 위해서 각 서버는 다음 두 기능을 수행해야 한다. 하나는 코루틴(coroutine) 기능으로서 순차처리형 프로그래밍 언어로 병존하는 여러 개의 프로세스를 모사하는 데 필요하다. 코루틴[7] 기능은 각 서버의 프로세스를 단위작업(activity) 단위로 관리하는 *ProcessBody(int)*와 서버보다 상위수준에 존재하며 공통의 시계(common clock)에 의하여 각 서버의 작업 순서를 결정하고 작업 시작을 알려주는 캘린더(calendar)간의 상호작용에 의하여 구현된다. 다음사건(next event)방식의 시뮬레이

션 스케쥴링 기법은 코루틴 사용의 전형적인 예이다. 프로세스간 상호작용방식의 구현을 위해 서버에 필요한 다른 하나의 기능은 통신기능이다. K-SIM 에서는 서버간의 통신을 위하여 *Signal(CMessage*)* 함수를 서버에 정의하였다. *Signal(CMessage*)*함수를 호출서버는 *CMessage* 클래스에 피호출서버를 호출한 목적과 호출서버 자신에 대한 정보, 그리고 작업물을 전달하고자 하는 경우에는 작업물에 대한 정보 등을 담아 전달한다. *Signal(CMessage*)*함수를 호출받은 서버는 메시지에 담긴 정보와 자신의 상태에 따라 적절한 반응을 취한다. 적절한 반응을 위한 일련의 과정이 *Signal(CMessage*)* 함수에 정의되어 있다.

3.3 K-SIM 의 적용사례

K-SIM 의 모델링 능력을 확인하기 위하여 G7 과제의 일환으로 개발된 첨단생산시스템과 T 사에서 개발한 교육용 FMS 의 모델링에 K-SIM 을 적용하였다. 첨단생산시스템은 4 대의 기계와 3 대의 로딩/언로딩 스테이션, 중앙스토커, 물류운반을 담당하는 스태커크레인, 공구운반시스템 등으로 구성되어 있다. 모델링 편의상 가공작업물을 3 종류로, 각 작업물의 가공에 필요한 공구도 2 가지로 단순화하였다. 교육용 FMS 는 2 대의 가공시스템과 AGV, 가공시스템과 AGV 사이의 작업물 운반을 담당하는 로보트, 그리고 중앙스토커와 스태커크래인, 스태커크래인과 AGV 사이의 물류를 담당하는 컨베이어 등으로 구성되어 있다. 위의 두 대상 시스템을 K-SIM 을 사용하여 각각 모델링한 후(부록의 <그림 2>와 <그림 3> 참조), 장비의 가동률과 생산량(throughput) 등의 통계량을 수집하였다. K-SIM 을 실제 문제에 적용한 결과 애니메이션 속도가 중요한 문제점으로 지적되었다.

4. 결론 및 향후연구과제

K-SIM 은 제조시스템용 시뮬레이터의 프로토타입으로서 시뮬레이션에 필요한 사전지식이 없는 제조시스템 현장의 FA 엔지니어가 손쉽게 사용할 수 있도록 다양한 기능을 제공한다. 모델구축 시간의 단축을 위하여 아이콘바탕의 모델링 기능을 제공하고, 모델의 적합성 검증을 위하여 3 차원 실시간 애니매이션을 지원하며, 모델링 유연성 향상을 위하여 스크립트 언어를 구비하였다. 또한, 통계적 지식이 없이도 시뮬레이터를 자유롭게 사용할 수 있도록 하기 위하여 입력데이타의 분포추정과 실행결과의 통계적 해석을 지원하는 통계분석기를 제공한다. K-SIM 의 다양한 기능을 제공하는 모듈을 효과적으로 개발하기 위하여 객체지향적 패러다임을 적용하여 SOOSM 을 이용한 프로세스간 상호작용방식의 시뮬레이션 엔진을 개발하였다. 본 연구에서 개발한 제조시스템용 시뮬레이터는 향후 가상제조시스템의 기본모듈로 사용할 수 있다.

K-SIM 의 기능 고도화를 위해서는 개발된 스크립트언어를 사용하여 사용자가 신규클래스를 정의하고 관리할 수 있어야 하며, 상태에 근거한 시스템 제어방식(state-based control)에 대한 연구와 이를 위한 효율적인 데이타구조에 대한 연구 등을 추후 과제로 생각할 수 있다.

참고문헌

[1] 이준성, 「자동화제조시스템 시뮬레이터의 운영규칙 모델링을 위한 스크립트 언어의 개발」, 한국과학기술원, 산업공학과 석사논문, 1997.

[2] 이진규, 「프로세스상호작용 방식의 자동화제조시스템 시뮬레이터」, 한국과학기술원, 산업공학과 석사논문, 1995.

[3] 이태억, 이진규, 임형규, 이진환, "자동화제조시스템의 객체지향적 모델링기법과 시뮬레이션구현방법: K-SIM 의 사례", 「IE Interface」, 8 권, 3 호(1995), pp. 47-60.

[4] 이태억, 임형규, 이진규, "프로세스상호작용에 의한 객체지향형 공장설계용 시뮬레이션 엔진의 개발", 「제 2 회 G7 첨단생산시스템 Workshop 논문집」, 1994, pp. 65-68 .

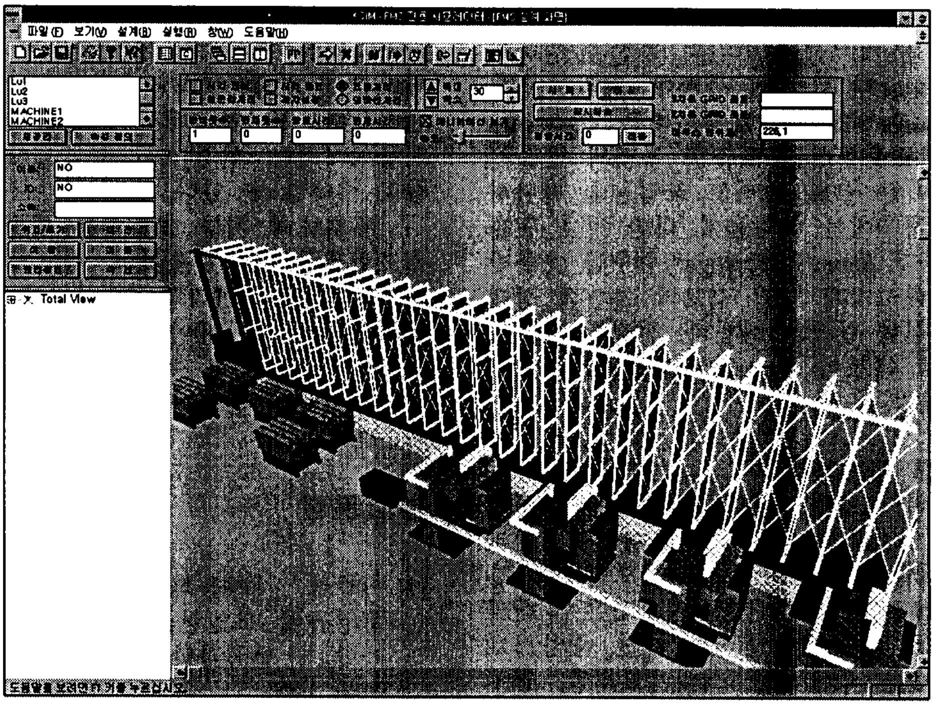
[5] Lee, J. S., Lee, J. K., Lee, J. H., Lee, T. E., and Oh, P. K., "Development of a Script Language for Modeling Operating Rules for an Automated Manufacturing System Simulator", 「'97 춘계공동학술대회 논문집」, 1997, pp. 782-785.

[6] Lee, T. E., and Lee, J. H., "A two-phase Approach for Design of Supervisory Controllers for Robot Cells: Model Checking and Markov Decision Models", Proceeding of IEEE Conference on Emergency Technologies and Factory Automation (also will appear in Annals of Operations Research), 1996, pp56-62.

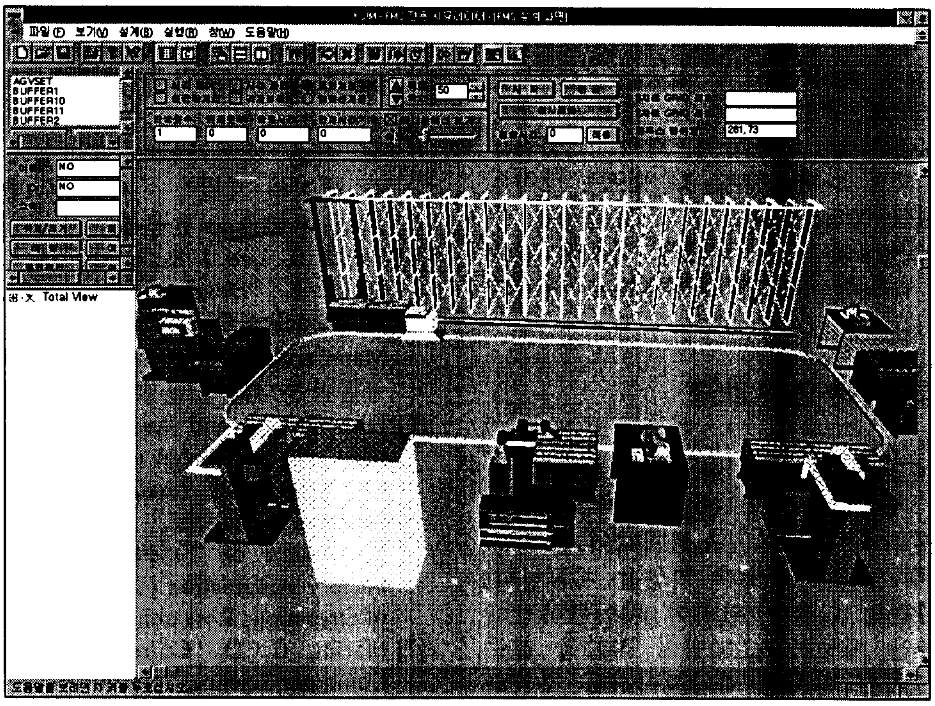
[7] Marlin, C. D., Coroutine, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1980.

[8] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., and Forenson, W., Object-oriented Modeling and Design, Prentice-Hall, 1991.

<부 록>



<그림 2> K-SIM 으로 구현한 첨단생산시스템 시뮬레이션 모델



<그림 3> K-SIM 으로 구현한 T 사의 교육용 FMS 시뮬레이션 모델