

모델 접속 기법에 의한 로봇 응용 생산시스템의 파라메트릭 시뮬레이션모델 개발

국 금 환*

Development of a Parametric Simulation Model by a Model Integration Method for Production System with Robots

Kum-Hoan Kuk*

ABSTRACT

In this study, a model integration method is presented as a new method for development of a parametric simulation model. This method enable us to integrate the special simulation models for each production subsystem into a large simulation model. Not only this large simulation model but also each special simulation model for each production subsystem can be used independently. Using this integration method man can reduce the development time and cost for simulation model development. To show the usefulness of this method, a simulation model for a production system with robots is developed by this model integration method. This simulation model is realized by the integration of two special simulation models, one model for a machining subsystem and the other model for a transport subsystem. The modeled production system consists of the robotic cells for machining and a transport subsystem which enable the material flow among the robotic cells. The flow of workpiece in each robotic cell is not fixed. All machines in a robotic cell are only served by robots.

Key words : Parametric Simulation Model, Simulation Program, Model Integration Method, Event, Robotic Cell, Production System, Production Subsystem

I. 서 론

생산 시스템의 자동화율이 높아지고 생산 시스템의 복잡도와 규모가 커짐에 따라 생산 시스템의 시뮬레이션

모델에 대한 요구도 변화 되었다. 즉 보다 정도가 높은 모델링 능력과 생산부 시스템들이 상호 긴밀이 연결되어 만들어진 대규모 생산 시스템의 모델링 능력이 요구되고 있다.⁽¹⁻³⁾ 또한 자동화율이 높은 생산 시스템의

* 경상대학교 공과대학 제어계측공학과(정회원)

경우 모델링을 통해 그 기능이 충분히 평가된 제어 규칙을 그대로 생산 시스템의 제어 프로그램 내에 구현하는 것도 요구되고 있다.⁽⁴⁻⁵⁾ 이와같은 생산 시스템 모델의 고정도화와 대규모화는 종국적으로 시뮬레이션 모델 개발 비용과 개발 기간의 증대를 의미 함으로 타 소프트웨어 제품 개발과 마찬가지로 시뮬레이션 모델의 효율적 개발을 위한 실용적 방법이 크게 요구 되며 지금도 계속 연구 되고 있는 실정이다.⁽⁶⁻⁹⁾

한편 최근의 자동화 추세중 하나는 로봇 셀(robotic cell)의 형성이며, 이러한 셀 내에서 산업용 로봇은 이송과 반송 작업을 위한 중요한 결합 요소로서 점차 응용 빈도가 커지고 있다. 이제껏 응용된 로봇은 주로 고정된 작업 순서에 따라 작업해 왔다.⁽¹⁰⁻¹¹⁾ 최근 상품 시장의 요구, 즉 다품종 소량 생산에 따라 로봇 셀이 갖는 유연성의 중요성이 커지게 되었다.⁽⁵⁾⁽¹²⁾ 이러한 셀의 특징은 로봇 작업 순서의 가변성이다. 이러한 고가의 로봇 셀 도입에 따른 투자 위험도를 줄이고 로봇 셀의 점진적 도입에 의하여 전 생산 시스템을 구성하기 위해서는 한 셀 내의 복잡한 관계뿐만 아니라 셀들 사이의 상호 관계도 정확히 시뮬레이션 할 수 있는 툴(tool)이 요구된다.⁽¹³⁻¹⁴⁾ 이러한 요구를 만족시킬 수 있는 시뮬레이션 시스템이 로봇 이용자와 로봇 응용 생산 시스템 설계자를 위하여 개발되어야 하며, 가급적 개발한 시뮬레이션 시스템의 이용도를 높이기 위해 이 프로그램 시스템이 개인용 컴퓨터(personal computer) 상에서 이용될 수 있어야 한다. 또한 작성된 프로그램 모듈들을 로봇 셀의 제어용 툴(tool)로서 손쉽게 이용할 수 있도록 미리 프로그램 설계시 고려해 주어야 한다.

본 논문의 목적은 파라메트릭 시뮬레이션 프로그램의 새로운 개발 기법으로서 시뮬레이션 프로그램 접속 기법을 제안하고 이 제안된 기법으로 작성된 하나의 실용적인 시뮬레이션 시스템을 보이는 것이다. 이 시뮬레이션 시스템은 로봇 셀의 설계와 제어를 위한 것으로 가변의 로봇 작업 순서를 갖는 한 셀 내의 동적 진행 뿐 아니라 로봇 셀들 사이의 상호 관계도 시뮬레이션 할 수 있다.

II. 본 론

II-1. 정 의

본 논문에 사용된 주요 개념을 다음과 같이 정의 한다.

- 시뮬레이션(simulation): 하나의 동적 현상에 대

한 지식을 얻기 위하여 이 동적 현상을 하나의 모델(model)로 나타내는 것.

- 시뮬레이션 프로그램(simulation program): 시뮬레이션 프로그램은 명령어들로 만들어진 컴퓨터 프로그램으로서 하나의 동적 현상의 시뮬레이션을 위한 독립된 모델(시뮬레이션 모델) 이고, 파라메트릭 시뮬레이션 프로그램은 특정 대상 시스템의 모델을 모델 파라미터로 정의 해주는 시뮬레이션 프로그램을 말하며 논문내의 시뮬레이션 프로그램은 파라메트릭 시뮬레이션 프로그램을 의미함.
- 시뮬레이션 시스템(simulation system): 시뮬레이션 시스템은 프로그램들의 집합으로서 적어도 하나의 시뮬레이션 프로그램과 데이터 입출력, 타 소프트웨어와의 접속등을 위한 보조 프로그램들로 이루어짐.
- 로봇 셀(robotic cell): 로봇 셀은 로봇을 가진 자동화된 가공 셀로서 몇개의 가공 기계와 버퍼 그리고 이것들을 상호 연결해주는 로봇으로 구성되며 가공 셀 내의 물류흐름을 제어는 셀 제어기가 담당함.
- 이동 로봇(moving robot): 로봇 본체의 장소 이동 기능을 가진 로봇.

II-2. 기존 모델 분석과 개발 사양 결정

II-2-1. 기존 모델 분석

본 논문의 목적에 관련된 시뮬레이션 프로그램들을 로봇과 물류 흐름 모델링 관점에서 분석하면 Fig 1 과 같이 아래 세 그룹으로 나눌 수 있다.

- 로봇 움직임의 모델링 정밀도는 매우 낮지만 물류 흐름 특성에 대한 많은 정보를 줄 수 있는 시뮬레이션 프로그램.⁽¹⁾⁽³⁾⁽⁴⁾⁽⁹⁾⁽¹⁵⁻²⁰⁾
- 로봇 움직임의 모델링 정밀도는 매우 높지만 물류 흐름 특성에 대한 정보는 거의 줄 수 없는 시뮬레이션 프로그램.⁽²¹⁻³⁰⁾
- 로봇 움직임의 모델링 정밀도도 약간 높고 물류 흐름 특성에 대한 정보도 많이 줄 수 있는 시뮬레이션 프로그램.⁽³¹⁻³³⁾

첫째 그룹의 시뮬레이션 프로그램의 경우 로봇 활동은 주로 하나의 부가적 시간, 즉 가공 준비 혹은 이송 시간으로 모델링 된다. 이와 같이 하나의 요소시간으로 로봇을 모델링하는 경우, 이러한 프로그램들은 다음과 같은 활용상 제한점을 갖는다.

		로봇 움직임의 모델링 정밀도	
		높다	낮다
비교할 수 있는 관련 프로그램에	고정	Program: G-GERT/31/, SIMDIS/32/ -----/33/	Program: SIMBED/1/, RENVIS/3/ SIMCON/4/, SOGS/9/ SIMULAP/15/, SIKTAS/16/ IMEDA/17/, DISC/18/ PROSS/19/, -----/20/
	이동	Program: GRIPPS/21/, CASOR/22/ ROBEX-4/23/, PROGRESS/24/ WADE-/25/, RIPE/28/ OSCAR/29/, -----/30/	

Fig. 1 Grouping of the related simulation programs

- 셀 내의 로봇 이송 순서가 가변인 경우, 로봇이 셀 요소들을 서비스 하는 서비스 규칙이 셀의 생산성에 미치는 영향을 알 수 없다.
- 타 생산 시스템 요소와 비교하여 로봇의 부하를 평가하는 것이 힘들다.
- 로봇의 고장이 물류 흐름에 주는 영향을 파악 할 수 없다.
- 시뮬레이션 프로그램을 셀의 제어용 프로그램으로 개조하는 비용이 매우 크다.

두번째 그룹의 프로그램들은 로봇이 한 셀이나 상호 연결된 셀들의 물류 흐름에 주는 영향에 대한 정보를 거의 줄 수 없다. 또한 세번째 그룹의 프로그램들은 단지 한 셀 내 고정 작업의 시뮬레이션을 위한 전용 프로그램 기능만을 갖는다. 본 논문에서 개발 하고자하는 시뮬레이션 프로그램은 세번째 그룹에 속하며 이 그룹에 속한 기존 프로그램들은 로봇의 이동성, 그리고 모델링, 로봇 이송 작업의 가변성, 로봇 서비스 규칙, 셀 내의 가공 규칙, 로봇 셀들 사이의 물류 흐름등의 모델링에는 제한성을 갖는다.

가변의 로봇 이송 작업 순서를 갖는 셀이 최근에 응용되었기 때문에 상기 기능을 비교적 정확히 시뮬레이션 할 수 있는 전용 프로그램이 아직 보고 되지 않은 상태이며 고정된 로봇 작업 순서를 갖는 셀의 설계용 시뮬레이션 프로그램들의 갯수도 적은데, 그 이유는 첫째 두 프로그램 그룹, 즉 로봇 셀 내에서의 충돌을 피하면서 주로 로봇의 작업 경로를 검토하기 위한 프로그램그룹과 생산 시스템 내의 물류 흐름을 시뮬레이션 하기 위한 프로그램 그룹의 상호 독립적 개발과, 둘째 타 생산 시스템 요소에 비해 로봇의 응용 기간이 짧은 점을 들 수 있다.

II-2-2. 개발 사양 결정

기존 관련 시뮬레이션 프로그램들의 분석과 기술 현

황으로부터 결정된 개발할 시뮬레이션 시스템의 사양은 다음과 같다.

(1) 시뮬레이션 시스템에 대한 상위 사양

물류 흐름 시뮬레이션 프로그램들의 개발 추이에 의하면 시뮬레이션 프로그램의 개발과 유지 비용이 과대한 점은 아직도 해결 해야할 문제점들 중 하나이다. 이 문제점에 대한 이유중 하나는 지금까지 프로그램 개발 시 시뮬레이션 프로그램의 기능성과 사용자 편의성 에 비해 프로그램의 장기적 확장 가능성을 등한시한 점들을 수 있다. 생산 시스템의 설계와 제어를 위한 소프트웨어로서 시뮬레이션 프로그램의 중요성이 커지고 생산 시스템 도입과 운영시 소프트웨어가 차지하는 비용의 비율이 커짐에 따라 시뮬레이션 프로그램의 장기적 확장 가능성을 개선 하기 위하여 아래의 상위 요구 조건을 결정 하였다.

- 규모가 큰 시뮬레이션 시스템을 개인용 소형 컴퓨터(PC)상에서 개발하는 방법의 안출.
- 안출한 시뮬레이션 시스템 개발 방법을 효율적으로 구현시킬 수 있는 설계 기준 작성.

(2) 시뮬레이션 시스템의 기능 사양

기존 관련 시뮬레이션 프로그램들과 기 도입된 로봇 셀들의 기능을 분석한 후 다음과 같은 사양을 결정 하였다.

- 생산 시스템의 작업 수행부 혹은 물류 흐름과 직접 관계를 갖는 생산 시스템 요소들의 모델링
- 로봇 셀과 로봇 셀, 로봇 셀과 반송 시스템의 상호 관계.
- 로봇 셀 요소 사이의 상호 관계, 즉 가공 기계, 버퍼 와 로봇 사이의 관계.
- 단일 혹은 이중 그리퍼를 지닌 고정 혹은 이동 로봇의 가변 이송 작업.
- 한 셀 내에서 상호 상이한 타입 가공물의 동시 가공.
- 한 셀 내에서 동종 가공물의 복수 가공 기계 상의 동시 가공.
- 생산 시스템의 제어 규칙 모델링
- 로봇이 서비스 할 셀 요소와 로봇을 선정하는 규칙.
- 셀 요소들과 운반 장치의 고장 발생 및 가공물을 셀에 입력하는 규칙.
- 로봇과 운반 장치가 다음 작업전에 대기 위치로 이동하는 규칙.
- 운반 요구를 운반 장치에 할당하는 규칙.

- 운반 경로 상에서 최적 운반 경로 결정 및 경로 진행 규칙.
- (3) 시물레이션 시스템의 설계 사양
- 데이터 구조 결정 및 모듈 분할을 위한 사양
 - 로봇 셀들로 구성된 다양한 생산 시스템을 통일적으로 기술하는 데이터 구조 구현.
 - 생산 시스템 요소 데이터와 가공 과제 데이터를 분리 하여 구현.
 - 소요 메모리의 최소화화 추후 데이터 구조의 확장성을 고려.
 - 모듈 분할시 모듈 사이의 접속(interface)을 단 순화 하고 상호 독립적인 컴파일(compile) 가능 성을 구현.
 - 시물레이션 프로그램의 흐름을 단순하게 파악 할 수 있도록 모듈 분할.
 - 타 소프트웨어와 직접적 데이터 교환 기능 구현.
- 시물레이션 시스템을 제어용 소프트웨어로 사용하기 위한 사양
 - 사상(event) 순서 제어 모델부와 사상 수행 모델 부의 명확한 분할.
 - 사상 수행 모델부의 시스템 상태 모델링을 정도 높게 구현.
 - 로봇 셀 상태와 제어 규칙 모델링 정밀도 변경의 유연성을 구현.

II-3. 개발 방법 안출

II-3-1. 시물레이션 시스템의 개발 방법

상호 연결된 로봇 셀들의 시물레이션 시스템을 개발 하기 위한 방법으로 아래 두가지 방법을 검토 하였다.

- 시스템 요소 위주의 방법

생산 시스템 요소가 생산 시스템의 특정 부시스템(subsystem)에 소속되는 소속성을 고려하지 않고 로봇 셀들(즉 부품 가공 부시스템)과 이들을 연결 해 주는 운반 부시스템에 대한 하나의 일반 시물레 이션 프로그램을 개발하는 방법.
- 부시스템 위주의 방법

각 부시스템들(즉 부품가공, 조립, 운반과 저장)에 대하여 각각 독자적으로 이용될 수 있는 전용 시물레이션 프로그램들을 개발하고 이들을 인터페 이스 모듈을 이용하여 상호 결합시키는 방법.

첫째 방법은 지금까지 가장 많이 사용해 온 방법이 고, 둘째 방법은 본 논문에서 새로 제안한 방법이다.⁽³⁴⁾

평가 항목		시스템 요소 위주의 방법	부시스템 위주의 방법
프로그 램	각 부시스템에 요구되는 모델링 정도의 개별적 만족 용이성	0	●
	시물레이션 시스템을 구성하는 각 모듈 들의 독립적 작성 용이성	0	●
개발	프로그래밍 작업의 규모	●	0
	부시스템들 사이의 관계 분석 용이성	0	●
프로그 램	상이한 부시스템 내 요소들 사이의 관계 분석의 용이성	●	0
	로봇 셀의 순차 도입시 소요되는 시물레이션 입력 데이터 준비 작업 규모	0	●
이 용	프로그램 크기를 주 메모리의 크기에 맞출 수 있는 가능성	0	●
	프로그램을 각 부시스템의 제어용 프로그램으로 변경 시릴 수 있는 용이성	0	●

평가(●: +, 0: 0, 0: -)

Fig 2. Comparison of methods for simulation model development

이 두가지 개발 방법에 대한 장단점을 Fig. 2에 정리 하였다. 전체적으로 부시스템 위주의 방법이 더 우수하 고 특히 이용 측면에서 우수하다. 즉 시물레이션 시스 템의 크기를 사용하는 컴퓨터의 주 메모리 크기에 손쉽게 맞출 수 있다. 또한 이 프로그램 결합 방법은 전용 프로그램의 장점과 일반 프로그램의 장점을 동시에 만족 시킬 수 있는 가능성과 시물레이션 프로그램을 설계 용도와 제어 용도로 개발할 때 발생하는 상호의 억제 작용을 동시에 감소 시킬 수 있는 가능성을 준다.

제안한 부시스템 위주의 방법에 의하여 상호 연결된 로 봇 셀들에 대한 시물레이션시스템을 개발 하기 위해서는 Fig. 3과 같이 가공 부시스템과 운반 부시스템에대 한 각각 독자적으로 수행 가능한 전용 프로그램과 이 두 프로그램의 상호 결합시 요구되는 인터페이스 모듈 이 개발 되어야 한다.

지금까지 시물레이션 프로그램들의 상호 결합이 등한 시 되어온 이유는, 각 부시스템에 대한 전용 시물레 이션 프로그램들이 독립적으로 개발되어 온 점과 부시스 템의 최적화에 비해 전 생산 시스템의 최적화는 비교적 최근에 관심을 갖게된 점을 들 수 있다.

Fig. 3의 방법으로 개발된 시물레이션 시스템에 의한 실제 생산 시스템의 설계 절차는 다음과 같이 수행 된다.

- 가공 부시스템 전용 시물레이션 프로그램에 의한 각 로봇 셀의 설계 (다양한 가공 과제들을 이용).
- 운반 부시스템 전용 시물레이션 프로그램에 의한 로봇 셀들의 연결 시스템으로서 운반 시스템의 설 계 (다양한 운반 과제들을 이용).

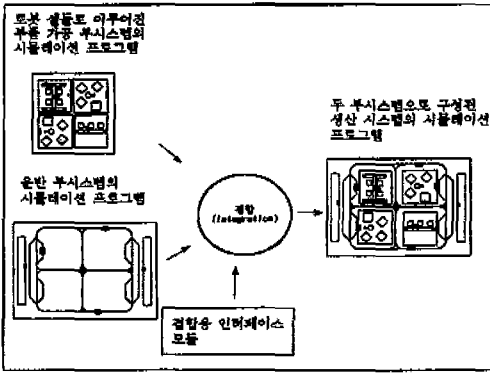


Fig. 3 Integration of two special simulation models independently used

- 두 전용 시뮬레이션 시스템을 상호 결합한 후 이 결합된 시스템을 이용하여 운반 부시스템을 통해 상호 연결된 전체 로봇 셀 시스템의 최적화.

II-3-2. 독립된 프로그램들의 상호 접속을 위한 통일화
 각 전용 시뮬레이션 프로그램은 독립적으로 개발하고, 또한 개발한 전용 프로그램들의 상호 결합이 가능하도록 아래와 같은 통일적 설계 기준을 작성 하였다.

(1) 프로그램 설계에 관한 통일화

프로그램 설계 단계에서 지켜야 하는 통일화 내용은 프로그램 개발 작업 순서에 따라 Fig. 4와 같이 작성 하였다.

(2) 프로그램 흐름에 관한 통일화

전용 프로그램들을 상호 결합 시키기 위한 Fig. 5의 시뮬레이션 프로그램 흐름에 관한 통일화 내용은 각 전용 프로그램 개발시 반드시 지켜져야 한다. Fig. 5에서 시뮬레이션 시계의 시각을 결정하기 위하여 다음과 같은 이유로 사상(event) 중심 시간 진행 방법을 결정하였다.

- 시간의 진행에 따라서 발생하는 사상들에 의한 시스템 상태 변화를 정도 높게 모델링 할 수 있으며, 이것은 작성한 프로그램을 시스템 제어용으로 사용하기 위한 전체 조건임.
- 페트리 넷트(Petri-Net)상의 사상 발생 시각과 실제 시스템의 사상 발생 시각이 잘 일치함.
- 소형 개인용 컴퓨터 성능이 급진적으로 좋아짐.

II-4. 시뮬레이션 시스템의 개발

본 절에는 II-3-2 절의 프로그램 설계에 관한 통일화 내용을 구체적으로 전용 시뮬레이션 프로그램, 즉 부품

설계 단계	통일화
관계 분석	- 시스템 요소 분석 - 시스템 요소들 사이의 관계 분석
	- 시스템 요소 속성의 분류 특성 - 관계들의 분류 특성 - 시스템 요소들 사이의 관계를 표현 하는 방법
모델 설계	- 시스템 동작과 제어 규칙 분석
	- 사상 순서의 기술 방법 - 제어 규칙의 분류 특성
	- 데이터 저장 구조 결정용 경로의 분류 특성 - 프로그램 모듈의 분할 방법 - 프로그램 입출력 데이터의 입출력 방법과 형식(format) - 마이크로컴퓨터와 접속을 위한 모델 데이터 교환 형식(format)

Fig. 4 Unification for program design

방 법	적용 방법
시뮬레이션 시간흐름 제어 방법	사상 중심 시간흐름 제어
시뮬레이션 진행중 사상 관리 방법	전체 사상 검색법
시뮬레이션 초기의 안정화 소요 시간 결정 방법	가변 시간 설정
시뮬레이션 진행중의 시스템 상태 확인 방법	중간 인쇄
시뮬레이션 진행 중절 방법	중 시뮬레이션 시간 설정

Fig. 5 Unification for program flow

가공 부시스템과 운반 부시스템의 전용 프로그램 개발 시 적용한 내용을 기술 하였다.

II-4-1. 데이터 구조 결정

(1) 시스템 요소 속성의 모델링

시스템 요소, 즉 계속 시스템 내에 머무르는 영구 요소와 일정 기간 동안만 시스템 내에 머무르는 일시 요소들이 갖는 모든 고유 속성은 시뮬레이션 프로그램 내부에서 다음 Fig. 6과 같이 정적, 동적, 통계적 속성으로 분류된다.

각 요소를 구분 혹은 결정해주는 속성은 정적 속성에 속하고 요소들의 동적 속성은 시뮬레이션 진행중 사상을 통하여 계속 변한다. 그리고 요소들의 통계적 속성은 개발하는 시뮬레이션 프로그램의 용도에 의해 결정되며 이 속성은 요소의 정적 속성과 동적 속성을 이용하여 정해진다. 시뮬레이션 프로그램의 모델링 정밀도는 사상의 정의와 요소 속성의 정의에 직접 의존한다.

(2) 생산 시스템 내에 존재하는 관계

시스템 요소사이의 관계는 시스템 요소간에 형성되는 시스템 고유 특성으로서 시스템 요소의 상태 변화에 의

분류 특성		도봇 속성의 예
제 조	- 시스템 요소의 타입과 변형 - 고정된 영구 요소의 위치와 - 임시 요소의 가공 관련 데이터	- 고유 번호 - 그리퍼 타입 - 가공할 가공 기계에 장탈착 가능한 순간 소요 시간 - 가공물을 버퍼에 장탈착 하는 평균 소요 시간
제 료	- 요소의 상태 - 임시 요소와 이동 - 영구 요소의 위치 - 사상 발생 시각	- 이동 작업 가능 여부 - 이동 중인 가공물 고유 번호 - 이동 경로 상의 위치 - 특정한 시스템 요소의 서비스 작업 완료 시각 - 고정 시작 시간 - 고정 보수 완료 시각
제 거	- 시스템 영구 요소의 불동과 이동 - 임시 요소의 가공 관련 상황	- 이동한 가공물 총 개수 - 총 서비스 시간 등(주행)한 총 거리와 시간 - 고정영역 외한 총 정지 시간 - 이동물

Fig. 6 Grouping criteria for attributes of system elements

존하지 않는다. 이러한 관계들은 Fig. 7과 같이 정리될 수 있으며 이 관계들이 생산 시스템의 구조를 나타내므로 이 관계들 자체가 한 생산 시스템에 대한 모델의 일부가 된다.

또한 실제 시스템의 복잡도에 비례하여 이러한 관계들의 갯수는 증가한다.

상기 시스템 요소사이 관계 외의 다른 두가지 관계를 생산 시스템의 동적 흐름으로부터 Fig. 8과 같이 이끌어 낼 수 있다.

(3) 시스템 요소들 사이의 관계 모델링

지금까지 시뮬레이션 프로그램의 데이터 구조 설계를 위하여 시뮬레이션 대상 내의 관계들을 가지적으로 표현하는 방법은 이용되지 않았다. 그러나 시뮬레이션 대상이 복잡해지고 규모가 커짐에 따라 시스템 요소 자체의 모델링 보다 시스템 요소들 사이의 관계를 모델링하는 것이 더욱 중요하게 되었다. 시스템 요소들 사이의 관계를 체계적으로 파악하고 검토하기 위하여 본 논문에서는 수정된 엔터티-관계 모델(Entity/Relationship-Model)을 보조 도구로 이용하였다.

이러한 모델링 결과를 프로그램 언어로 표현하기 위하여 본 논문에서는 PASCAL 언어를 사용 하였다. PASCAL 과 C 는 모두 동적 메모리 관리 능력을 가지므로 소모 메모리 측면에서는 둘다 우수 하지만 PASCAL은 프로그램 해독성 측면에서 보다 우수하다.

II-4-2. 프로그램 구조 결정

(1) 사상 흐름의 모델링

불연속 시스템의 동작은 임의 시각에 발생한 사상들에 명확히 모델링 될 수 있고, 시스템의 동작 제어는

분류 특성			예	
			R1	R2
시스템 요소의 종류	영구 요소 사이의 관계	하나의 생산 부시스템에 속한 요소간의 관계	0	0
		서로 다른 생산 부시스템에 속한 요소간의 관계		0
	영구 요소와 임시 요소 사이의 관계			
기술적 특성	순수한 공간상의 거리 관계			0
	기술과 기능 관련 관계		0	

O : 빈공한
R1 : 하나의 도봇 셀 내의 가공 기계와 버퍼 사이의 관계
R2 : 도봇 셀들을 연결해 주는 운반 시스템 반송 경로 상의 마디(node)와 가지(arc)의 관계

Fig. 7 Grouping criteria for relations between system elements

종류	용도와 도출 경로	관계의 용도	관계로부터 이끌어낸 정보
시스템 요소들 사이의 관계		시뮬레이션 대상 시스템의 구조 모델링	대상 시스템 모델의 데이터 구조
시스템 요소의 상태들 사이의 관계		시스템 상태 변화의 모델링	각 시스템 요소들에 대한 사상(event)
시스템 요소들 상태 변화들 사이의 관계		시스템 작업 흐름의 모델링	시스템 작업 흐름의 제어 규칙의 흐름

Fig. 8 Type of relations between system elements

사상들의 스케줄링을 통하여 행해진다. 시뮬레이션 프로그램의 기능상의 신뢰도를 높이고 개발 작업 자체의 효율을 높이기 위하여 페트리 넷트(Petri-Net)를 사상 발생 순서의 모델링 도구로서 채택 하였다.

페트리 넷트를 이용하여 시뮬레이션 대상 내의 동적 거동을 검토할 수 있고, 동시에 시뮬레이션 대상이 갖는 기본 상태와 기본 사상 및 시스템 요소들의 공통 천이에 의한 상호 접속등을 이끌어 낼 수 있다. 페트리 넷트로 부터 얻어진 정보들은 다음과 같이 프로그램 개발의 각 주요 작업을 위한 핵심 정보가 된다.

- 기본 상태 → 사상 생성의 모델링
- 기본 사상 → 사상 수행의 모델링
- 시스템 요소들의 공통 천이에 의한 상호 접속 → 사상 선택의 모델링

특정 시뮬레이션 대상에 대한 기본 상태, 기본 사상과 시스템 요소들의 공통 천이에 의한 상호 접속의 정보를 얻기 위하여 페트리 넷트로 모델링할 실제 시스템 동작은 아래의 조건을 만족시켜야 한다.

- 시스템 동작이 시스템 요소들 사이의 실제적 의미를 갖는 가능한 관계들을 모두 포함.
- 시스템 동작이 시스템 경계에서의 가능한 모든 사

상을 포함.

(2) 시뮬레이션 시스템의 모듈 분할

모듈화는 문제의 복잡도를 줄이기 위한 하나의 프로그래밍 기법이다. 각 생산부시스템에 대한 독립적인 시뮬레이션 모델을 작성하기 위하여 각 부시스템의 사상 순서 제어 작업은 Fig. 9의 계층 2에 상호 독립적인 고유 모듈로 구현 시켰다. 각 생산 부시스템에 대한 전용 시뮬레이션 프로그램들이 독립적으로 활용 될 때는 계층 1의 모듈은 요구되지 않지만 적어도 두개 이상 전용 프로그램들의 상호 결합이 요구 될 때는 계층 1의 모듈이 요구 된다. 사상 순서 제어 모듈들을 계층 1과 2로 나누어 실현 함으로서 기존의 모듈 라이브러리 개념이 프로그램 라이브러리 개념으로 확장 되었으며 이것을 통해 전용 시뮬레이션 프로그램들의 상호 접속 작업이 매우 단순하게 실현 되었다. 또한 계층 3의 모듈은 새로운 제어 규칙의 추가나 모델 정밀도 수정 작업이 용이 하도록 모듈분할을 수행 하였다. 이를 통해 작성한 모듈을 실시간 제어용 프로그램으로 활용하기 위한 추가 작업이 단순해 진다. 시뮬레이션 데이터 입력 모듈과 시뮬레이션 결과 출력 모듈을 독립적인 모듈로 분리 구현함으로서 필요시 프로그램 사용자의 데이터 입출력

방식 변경 요구를 신속히 만족 시킬 수 있다.

II-4-3. 제어 규칙 결정

시뮬레이션 프로그램 속의 제어 규칙은 실 시스템 운전시 요구되는 의사 결정기능을 모델링한 것이며 다음 두 가지 내용을 통하여 정의 될 수 있다.

- 기능.
 - 제어 규칙의 적용 조건으로서의 시스템 상태.
- 실 시스템 내 의사 결정의 제어 규칙에 의한 모델링 작업은 실 시스템 내부의 물류 흐름에 대한 정보가 시간적 지연없이 얻어진다는 이상 조건을 가정한 것이다.

(1) 부품 가공 로봇 셀의 제어 규칙

로봇 셀의 페트리 넷으로부터 얻은 사상들(공통 천이)에 의한 셀 요소들의 결합과 셀 경계에서의 사상에 참여하는 셀 요소들은 다음 6 개의 제어 규칙을 통하여 결정 된다.

- 로봇이 서비스 해야 하는 셀 작업 요소의 결정(선착순:FIFO, 최소 가공 시간:SMT, 최대 가공 시간:LMT).
- 서비스 할 로봇의 결정(선착순).
- 버퍼로부터 출력할 가공물의 결정(선착순).

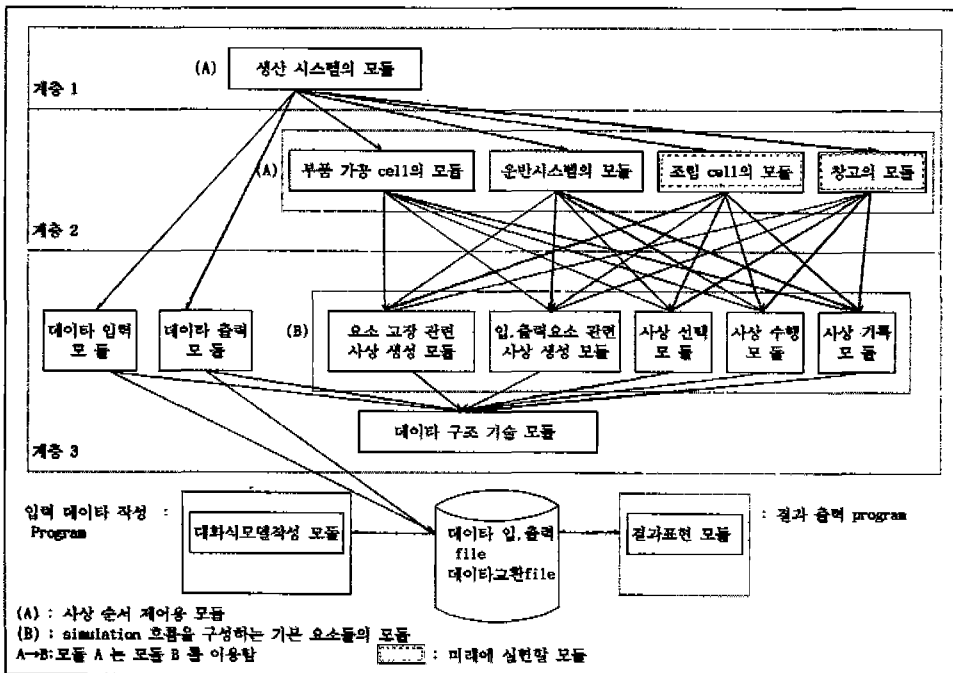


Fig. 9 Module division of simulation system

- 가공물(workpiece)을 로봇 셀에 입력(난수 발생기와 가공물 타입당 생산 목표 수량, 개별 입력과 룯드 입력).
 - 셀 작업 요소들의 고장 발생(확정값, 통계값).
- (2) 운반 부시스템의 제어 규칙

다음의 제어 규칙들은 무인 반송 시스템에 요구 되는 제어 규칙 들이며 이경우 반송 경로상의 무인 반송차 제어가 가장 중요하다.

- 요구된 반송 작업들로 부터 수행할 반송 작업의 결정(선착순).
- 운반할 무인 반송차 결정(선착순, 무인 반송차 위치).
- 운반할 무인 반송차의 운반 경로 결정(Dijkstra 알고리즘, Floyd 알고리즘).
- 무인 반송차의 이동(실시간 반송 경로 상황 고려에 의한 블록 단위 경로 진행).
- 반송 작업 요구의 발생(난수 발생기, 실제 반송 작업 요구).

II-4-4. 시뮬레이션 프로그램의 접속과 흐름

(1) 전용 프로그램의 접속

두 전용 프로그램이 접속된 후 한 로봇 셀 내의 셀 요소는 두개의 번호, 즉 셀 번호와 셀 요소 번호로 구별된다. 프로그램 접속시 부품 가공 로봇 셀들과 운반 부시스템의 공간적 관계는 각 셀의 입출력 위치 번호와 반송 시스템 내의 마디 번호를 일치 시켜 줌으로서 정의된다. 만약 반송 경로가 한 로봇 셀 내부까지 들어오는 경우는, 즉 반송 시스템과 가공 셀의 상호 중첩은 반송 시스템의 마디 번호와 셀 요소 번호를 일치시켜 나타낸다.

복수 개의 로봇 셀들이 하나의 반송 시스템에 의해 상호 연결되면 로봇 셀과 반송 시스템 사이에 끊임없는 상호 작용이 존재한다. 이러한 상호 작용은 하나의 일시 요소가 로봇 셀로부터 반송 시스템으로 들어가거나 반송 시스템으로부터 로봇 셀로 들어가는 것을 의미한다. 가령 가공물로 가득 찬 팔레트가 셀 출력 버퍼상에 존재 하거나 셀 입력 버퍼상에 팔레트를 위한 빈 공간이 발생 하면 새로운 반송 작업 생성이 검토된다. 만약 이러한 검토 결과가 긍정적이면 이 새로운 반송 작업 요구는 반송 작업 요구 리스트에 등록 된다.

이러한 두 부시스템 사이의 상호 작용은 시뮬레이션 진행중 인터페이스 모듈과 두 부시스템 고유의 사상 수행

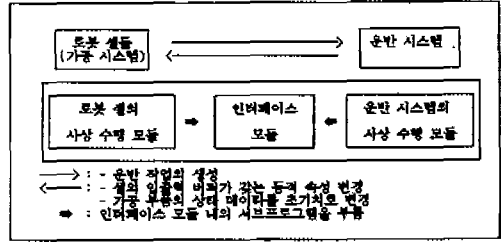


Fig. 10 Interaction between the machining system and the transport system

모듈을 약간 수정 함으로서 모델링 된다. 이 인터페이스 모듈 내에 구현한 서브프로그램들은 Fig. 10과 같이 사상 수행모듈로 부터 직접 호출된다.

(2) 접속된 프로그램의 작업 흐름

Fig. 9의 모듈들 중에서 프로그램의 상호 접속에 의존하는 모듈은 사상 수행모듈과 사상 순서 제어 모듈이다. 이 2 종의 모듈을 각 생산 부시스템 전용의 독자 수행 프로그램의 한 모듈로 사용 할 때나 결합된 프로그램의 모듈로 사용 할 때 두 경우에 대하여 동일한 형태로 사용하기 위하여 이 두 모듈 내의 프로그램 상호 결합에 의존 하는 부분에 하나의 파라미터를 이용한 프로그램진행 제어 기능을 삽입 하였다.

이 파라미터에 의하여 결합후의 프로그램 흐름 제어가 이루어지고 각 프로그램 결합에 의존하는 모듈 내의 프로그램 흐름이 제어된다.

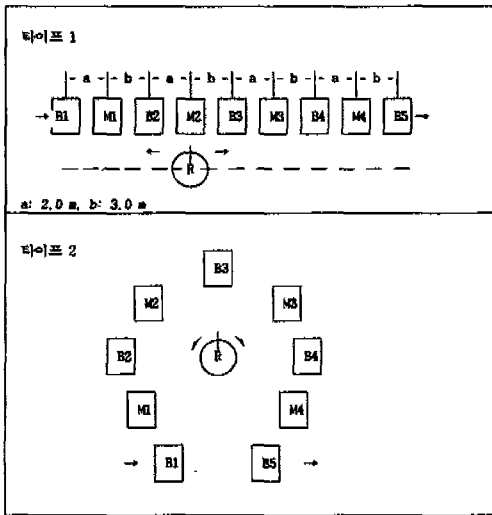
III. 개발 시스템의 적용

III-1. 하나의 로봇 셀에 적용

III-1-1. 시뮬레이션 대상

시뮬레이션 대상으로서 다음 Fig. 11와 같은 하나의 로봇 셀을 결정하였다. 두 타입의 각 셀에 대하여 첫째 버퍼는 셀 입력 버퍼이고 맨 끝의 버퍼는 셀 출력 버퍼이며 전배어 장치에 의한 입출력을 나타낸다. 그리고 n 번 버퍼는 n 번 가공 기계에 속한 입력 버퍼이다. 로봇은 셀 내 모든 가공 기계와 버퍼의 가공물 입출력 서비스 기능을 담당한다.

가공물(workpiece)은 시뮬레이션 프로그램에 내장된 난수 발생기에 의하여 결정된 다음 Fig. 12와 같은 네가지 타입이 셀에 입력된다. 가공물들의 평균 도착율(단위 시간당 도착한 가공물의 평균 갯수)과 가공물의 평균 가공 완료율(단위 시간당 가공 완료한 가공물



Bn: buffer number n, Mn: machine number n
R: robot (이중 그리퍼, 이동 속도: 1.0 m, 가공 기계의 가공물 입출력 시간: 2 s, 버퍼의 가공물 입출력 시간: 1 s)

Fig. 11 Layout of robotic cell

workpiece type	machining sequence and time (n/x)			
	M1	M2	M3	M4
1	1/63	2/62	3/65	-
2	-	1/71	2/35	3/104
3	1/99	-	2/57	3/54
4	1/61	2/73	-	3/66

Mn : machine number, n
n/x : (n: machining sequence, x: machining time (s))

Fig. 12 Machining sequence for each workpiece

의 평균 갯수)의 비율을 가공 흐름 밀도(ρ)라 정의하고, 시뮬레이션을 위하여 가공물들의 도착율은 포아송 분포로 평균 가공 완료 시간(1/가공물의 평균 가공 완료율)은 일정값(= 210 초)으로 정한후 가공 흐름 밀도(ρ)값을 0.8, 1.0, 1.2, 1.4, 1.6 으로 결정 하였다.

III-1-2. 시뮬레이션 결과

주어진 로봇 셀과 가공 과제에 대하여 로봇이 서비스해야하는 셀 작업 요소를 결정하는 우선 규칙(priority rule)의 영향을 시뮬레이션에 의하여 비교 하였다.

시뮬레이션 할 때 셀 요소의 고장과 가공 공구 교환은 고려 하지 않고 가공물은 개별적으로 입력하였다.

또한 시뮬레이션 초기 상태가 시뮬레이션 결과에 미치는 영향을 제거하기 위하여 몇번의 시뮬레이션 실험을 통해 초기 20000 초의 기간을 시뮬레이션 초기 안정화 시간값으로 결정하였다. 총 시뮬레이션 시간 80000 초에 대한 시뮬레이션 결과는 Fig 13과 같다.

시뮬레이션 결과에 의하면 선착순(FIFO) 규칙이 가공물 평균 가공 시간에 대하여 비교적 우수함을 알 수 있다. 또한 가공물 흐름 밀도값(ρ)이 커짐에 따라 셀 요소의 이용율은 높아지지만 가공물의 평균 가공 시간은 길어진다.

III-2. 상호 연결된 로봇 셀들에 적용

III-2-1. 시뮬레이션 대상

이곳에서는 두 개의 독립된 전용 시뮬레이션 프로그램을 상호 접속한 프로그램을 기존 생산 시스템의 변경(Fig 14)을 위한 시스템 설계용 도구로 적용한 경우를 보이고자 한다. 기존 생산 시스템(타이프 1)은 2개의 로봇 셀로 구성되고 각 셀내에서 부품 가공이 완전히 수행되므로 두 셀 사이의 물류 흐름은 존재하지 않는다. 이때 각 셀 내의 모든 셀 요소들은 각 1대의 이동 로봇에 의하여 서비스 된다. 그러나 셀 내에서의 완전한 가공을 위하여 두 셀이 동일한 특수 가공 기계를 가져야 하고 이로 인해 이 기계의 이용률은 매우 낮다.

이러한 문제점의 해결 방법으로 특수 가공 기계 한 대와 한 대의 로봇으로 구성된 제 3의 로봇 셀을 만들고 기존 셀로부터 특수 가공 기계들을 제거한 새로운 생산 시스템(타이프 2)이 제안 되었다. 여기에서 된 생산 시스템의 셀 1의 5번 기계와 셀 2의 4번 기계가 동일한 특수 기계이며 두 셀 내의 로봇은 이중 그리퍼를 가진 이동 로봇이다.

또한 재설계한 타입 2 생산 시스템의 경우 미래의 점진적 추가 셀 도입을 위하여 Fig. 15와 같이 셀들을 상호 연결 시켜주는 무인 운반 시스템을 도입 하였다. 상기 로봇 셀들에서 수행하는 가공 과제는 Fig. 16과 같은 4종의 가공 부품들을 가공하는 것이다. 가공 부품들의 입력은 타입 1과 3의 경우 가공 부품 10개 용량의 팔레트로 타입 2과 4의 경우 가공 부품 8개 용량의 팔레트로 각 셀에 입력된다.

타이프 1, 3 가공 부품의 평균 입력 간격은 5000초이고 타입 2, 4 가공 부품의 평균 입력 간격은 3000 초 이다. 그러나 각 입력 시각에 입력되는 가공 부품 타입은 난수 발생기에 의하여 결정된다.

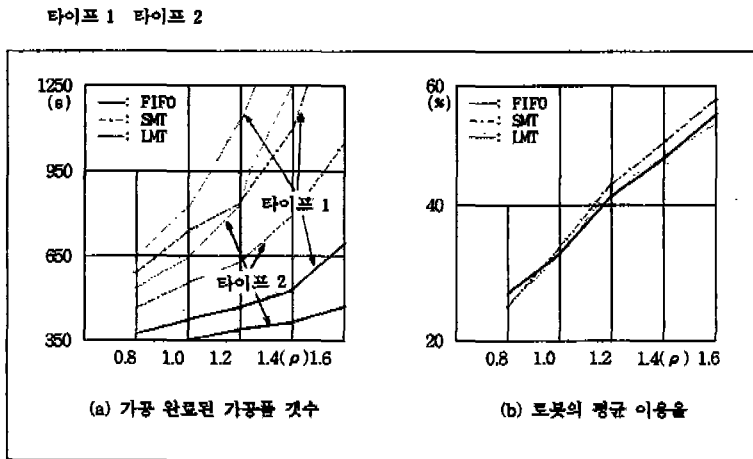


Fig. 13 Simulation results

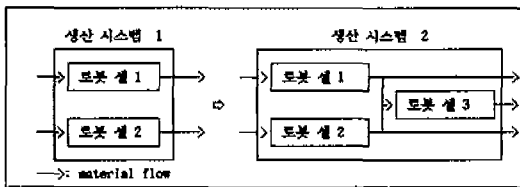


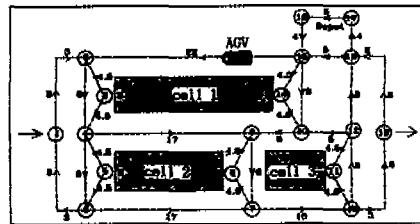
Fig. 14 Introducing of robotic cell 3

III-2-2. 시뮬레이션 결과

시뮬레이션 결과들 중 타이프 1, 2 두 생산 시스템 요소들에 대한 이용률은 Fig. 17과 같다. 그림에서 알 수 있듯이 로봇 셀 3의 도입으로 각 셀 내의 가공기계 이용률이 상호 비슷해지고 또한 약간씩 높아 졌다. 그러나 셀 3은 셀 1, 2의 상호 독립성을 낮추는 작용을 한다. 또한 셀 3의 로봇 이용률이 아주 낮으므로 필요 시 로봇 셀 3에 제 2의 공용 특수 가공 기계를 추가 설치 할 수 있음을 알 수 있다. 또한 무인 운반차의 이용률(42.8%)도 낮음으로 현 반송 시스템을 이용한 추가 셀 도입 역시 고려 할 수 있다.

IV. 결 론

시뮬레이션 프로그램의 새로운 개발 기법으로 본 논문에서 제안한 모델 접속 기법은 시뮬레이션 대상에 따라 요구되는 전용 시뮬레이션 프로그램의 상호 결합을 가능케 하였다. 이를 통해 시뮬레이션 대상 시스템의



→: 각 경로에 허용된 이동 방향
 B: cell input buffer E: cell output buffer
 n: node number n —⊕—: arc length l
 AGV: 무인 운반 차 (갯수:1, 평균 이동 속도: 1.0 m/s, 실는 시간: 30 s, 내리는 시간: 60 s)

Fig. 15 Transport route in production system 2

모델링 요구 정밀도를 개별적으로 만족 시킬 수 있으며 주어진 컴퓨터의 주 메모리 크기에 맞추어 프로그램 크기를 결정 할 수 있다. 즉 한 프로그램 내의 기존 모듈 개념을 프로그램 시스템 내의 모듈 개념으로 확장 하였다. 그러나 상호 독립적으로 수행되는 두 전용 프로그램이 접속되기 위해서는 두 전용 프로그램 개발시 제안한 프로그램 흐름에 대한 통일화 내용이 지켜져야 한다. 개발한 시뮬레이션 시스템은 파라메트릭 시뮬레이션 시스템(parametric simulation system) 이므로 시뮬레이션 대상은 모델 파라미터에 의하여 정의 된다. 이 새로운 개발 방법으로 작성된 시뮬레이션 시스템은 상호 연결된 로봇셀의 설계 작업을 위한 판단 정보를 제공할 뿐 아니라, 셀 내 이송 작업들의 스케줄링 도구가 될 수 있으므로 시뮬레이션을 통해 검증된 사상 선택 모듈 내의 제어 법칙들은 로봇 셀 제어용 프로그램

생산 시스템		가공물 타이프 번호	가공공정 상의 기계 수	가공 소요 시간 (s)				
				M1	M2	M3	M4	M5
1 비 타입	로봇셀 1	1 2	5 4	103 139	122 97	105 94	137 107	135
	로봇셀 2	3 4	3 4	98 112	115 105	142 132	127	- -
2 비 타입	로봇셀 1	1 2	4 4	103 139	122 97	105 94	137 107	- -
	로봇셀 2	3 4	3 3	98 112	115 105	142 132	- -	- -
	로봇셀 3	1 4	1 1	135 127	- -	- -	- -	- -

Mn : machine number n
 □ : machining at the special machine

Fig. 16 Machining sequence for each workpiece

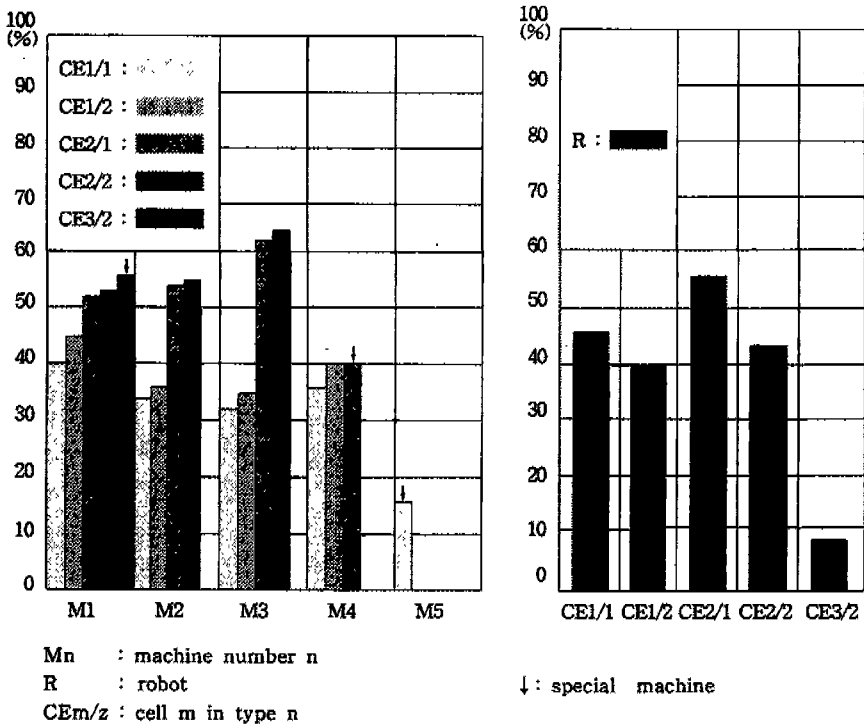


Fig. 17 Simulation results

의 주 부분으로 이용 될 수 있다.

엔터티-관계 모델을 이용한 데이터 구조 설계 와 페트리 네트를 이용한 사상수행의 모델링은 시뮬레이션 모델의 검증 과 가시화 모델 즉 실제 시스템과 시뮬레이션 모델 사이의 중간 모델로 보다 체계화하고 구체화한 후 차후 발표할 예정이다. 그리고 두 개의 생산 부 시스템에 대한 전용 시뮬레이션 프로그램을 완성 하였지만 나머지 두 개의 생산 부시스템에 대한 전용 시뮬레이션 프로그램을 작성 하는 것도 미래의 과제이다.

V. 참고 문헌

1. M. G. Ketcham, J. Watt, "Parametric simulation for flexible assembly systems", *Journal of Manufacturing Systems*, Vol. 8, No.2, pp.115-125, 1989.
2. I. Sabuncuoglu, D. L. Hommertzhaim, "Expert simulation systems- recent developments and applications in flexible manufacturing systems", *Computers ind. Engng*, Vol.16, No.4, pp.575-585, 1989.
3. R. M. O'Keefe, J. Haddock, "Data-driven generic simulators for flexible manufacturing systems", *Int. J. Prod. Res.*, Vol.29, No.9, pp.1795-1810, 1991.
4. A. Bilberg, L. Alting, "When simulation takes control", *Journal of Manufacturing Systems*, Vol.10, No.3, pp.179-193,1991.
5. L. Lin, R. Sanka, "Data srructure and process data traversal in a manufacturing cell controller", *Computers ind. Engng*, Vol.24, No.1, pp.23-32, 1993.
6. H. M. Su, V. Kachitvichyanukuk, "A natural langusge system to aid simulation model formulation", *Computers ind. Engng*, Vol.16, No.4, pp.535-543, 1989.
7. J. Haddock, N. Seshadri, V. R. Srivatsan, "A decision support system for simulation modeling", *Journal of Manufacturing Systems*, Vol.10, No.6, pp.484-491, 1991.
8. C. O. Benjamin, L. L. Massay, M. Sivapiromrat, "A winning strategy for managing simulation projects", *Computers and Industrial Engineering*, Vol.23, No.1-4, pp.229-232, 1992.
9. Y. Yuan, C. A. Dogan, G. L. Viegelahn, "A flexible simulation model generator", *Computers ind. Engng*, Vol.24, No.2, pp.165-175, 1993.
10. J. Milberg, "Entwicklungstendenzen in der automatisierten Produktion", *Technische Rundschau* 37, pp.42-43, 1985.
11. D. Schellenberger, H. Scheibner, W. Bahmann, H. G. Moldenhausner, "Moeglichkeiten und Trends bei der automatischen Werkstueckhandhabung mit Industrierobotern", *wt-z. ind. Fertig.*, 76, Nr. 10, pp.585-589, 1986.
12. G. Handke, "Der Roboter als CIM-Komponente", *Praktiker-Tagung: Prozessintegrierer Robotereinsatz*, Muenchen, 26-27 Maerz, 1987.
13. G. Pritschow, "Die flexible Fertigungszelle", *Fertigungstechnisches Kolloquium*, Stuttgart, pp.48-57, 10-12 Oktober, 1985.
14. S. H. Teng, J. T. Black, "Cellure manufacturing systems modeling: the Petri net approach", *Journal of Manufacturing Systems*, Vol.8, No.2, pp.115-125, 1989.
15. H. Steffens, "Ein Beitrag zur Optimierung der Prozessfuehrungsstrategien automatisierter Foerder-und Materialflusssysteme", *Diss. Uni. Stuttgart*, 1983.
16. S. Chmielnicki, "Simulation der Prozesse in flexiblen Fertigungs-systemen als Hilfsmittel zur Planung und zum Test von Steuerungsprogrammen", *Diss. Uni. Stuttgart*, 1985.
17. P. Floss, J. Talavage, "A knowledge-based design assistant for intelligent manufacturing systems", *Journal of Manufacturing Systems*, Vol.9, No.2, pp.87-102,

- 1990.
18. S. Selvaraj, E. L. Blair, M. L. Smith, W. M. Marcy, "Discrete event simulation in c with DISC", *Computers ind. Engng*, Vol.18, No.3, pp.263-274, 1990.
 19. J. Haddock, R. M. O'Keefe, "Using artificial intelligence to facilitate manufacturing systems simulation", *Computers ind. Engng*, Vol.18, No.3, pp.275-283, 1990.
 20. Y. S. Sherif, R. G. Dear, "A pascal simulation model for the analysis of queueing systems", *Computers ind. Engng*, Vol.20, No.3, pp.303-311, 1991.
 21. S. J. Azzam, M. U. Unuvar, "Off-line Robot Programming with GRIPPS", *ROBOTS-9*, Detroit Michigan, 1985.
 22. H. J. Warnecke, A. Altenhein, "Zwei Verfahren zur Kollisionserkennung und Vermeidung bei der Off-line Programmierung von Industrierobotern", *Robotersysteme 2*, pp.163-169, 1986.
 23. M. Weck, Th. Niehaus, M. Osterwinter, "Graphisch interaktives Programmier- und Testsystem fuer Industrieroboter", *Robotersysteme 2*, pp.193-201, 1986.
 24. A. Naylor, L. Shao, R. Volz, "PROGRESS- A graphical robot programming system", *IEEE*, pp.1282-1291, 1987.
 25. T. Inoue, H. Yasukuni, A. Nakazumi, T. Arai, A. Matsumoto, "Robot language system with motion simulator", *R&D Kobe Steel Engineering Reports*, Vol.37, No.2, 1987.
 26. J. A. Fernandez, F.A. Harrison, "3D robot simulation environments", *INCOM '89*, pp.455-460, 1989.
 27. A. Levas, R. Jayaraman, "WADE: an object-oriented environment for modeling and simulation of workcell applications", *IEEE Transactions on robotics and automation*, Vol.5, No.3, pp.324-336, 1989.
 28. D. J. Miller, R. C. Lennox, "An object-oriented environment for robot system architectures", *IEEE*, pp.352-361, 1990.
 29. U. Olschewski, "Flexibel automatisierte Montagebereiche offline programmieren", *VDI-Z 132*, Nr. 7, pp.60-65, 1990.
 30. T. Ueno, "Off-line programming of construction robots", *JRSL*, Vol.8, No.2, pp.71-75, 1990.
 31. D. J. Medeiros, R. P. Sadowski, "Simulation of robotic manufacturing cells: a modular approach", *SIMULATION*, January, pp.3-12, 1983.
 32. H. Zachau, A. Rebentrost, "Simulation-untersuchung einer flexiblen Fertigungszelle fuer die spanende Bearbeitung", *Fertigungstechnik und Betrieb*, Vol.34, Nr. 10, pp.615-619, 1984.
 33. E. Niemi, B. J. Davies, "Simulation of an optimizing FMS-cell control system", *Robotics & Computer-Integrated Manufacturing*, Vol.5, No. 2/3, pp.229-234, 1989.
 34. H. J. Warnecke, K. H. Kuk, H. S. Cho, "Modular Simulation Model of Robot Cells", *INCOM '89*, Madrid Spain, pp.461-465, 1989.