

부품상태기반 일정계획 시뮬레이터의 개발에 관한 연구

김기태*, 박찬권**, 박진우*

* : 서울대학교 산업공학과

** : 영산대학교 경영정보학과

A Study on the Development of Part State Driven Scheduling Simulator

Ki Tae Kim*, Chan Kwon Park** and Jin Woo Park*

* : Dept. of Industrial Engineering, Seoul National University

** : Dept. of Management Information, Young San University

요 약

고가의 생산장비를 효율적으로 운영하기 위한 방법을 제시하는 방법으로 일정계획은 중요한 역할을 담당한다. 일정계획을 수립하기 위한 방법으로 많은 방법이 적용되고 있지만 시뮬레이션은 일정계획의 수행도 분석에 중요한 기능을 담당한다. 그러나 일정계획의 분석에 있어서 적용되는 기존의 시뮬레이션 방법에는 대상 시스템을 모형화 하기 위해 적지 않은 노력이 필요하고, 사용하는 시뮬레이션 프로그램의 사용 방법 및 모형 작성 방법에 대한 학습이 필요한 경우도 적지 않다.

본 논문은 일정계획을 시뮬레이션하기 위한 일정계획 시뮬레이터 개발에 대한 연구이다. 기존의 시뮬레이션 프로그램이 가지는 이벤트 개념 대신 상태 개념을 이용하여 시뮬레이션을 진행하는 방법을 이용하였다. 시뮬레이션하고자 하는 시스템에 존재하는 주요한 객체들의 상태를 분석하여 시뮬레이터를 개발하는 방법을 제시하였다. 생산시스템에 대한 모형을 작성하고자 할 경우 특별한 프로그래밍 작업없이 대상시스템의 모형을 작성하도록 하여 보다 편리하게 모형을 작성하였다. 대상시스템에 대한 프로그램없이 대상시스템의 구성 정보와 주문 정보를 입력하여 우선순위규칙에 의한 다양한 일정계획의 수행도를 분석하는 일정계획 시뮬레이터를 개발하였다.

주요어 : 객체상태, 일정계획, 시뮬레이터, 우선순위규칙

1. 서론

기존의 시뮬레이션 프로그램은 이벤트의 형태에 따라서 정해져 있는 이벤트 루틴을 수행하였다. 시뮬레이션을 위하여 대상 시스템에서 발생할 수 있는 모든 종류의 이벤트를 선언하고 각 이벤트에 대응하는 자료의 수정을 정의하여 각 이벤트마다 대응하는 프로시저를 설정하여 시뮬레이션을 실행하였다.

이런 이유로 시뮬레이션 프로그램에서는 많은 종류의 이벤트가 정의될 필요가 있고 많은 이벤트는 해당 시뮬레이션 프로그램을 사용하고자 하는 시뮬레이션 분석가에게 부담으로 작용하였다. 이를 위해서 기존의 상용 시뮬레이션 프로그램은 사용자가 보다 쉽게 시뮬레이션 모형을 작성하도록 하는 방안을 제시하고 있다.

본 논문에서는 기존의 시뮬레이션 프로그램이 가지는 사건 기반 시뮬레이션 프로그램에 대하여 특정한 영역에 존재하는 객체의 상태에 대하여 시뮬레이션이 수행되도록 하는 부품상태기반 시뮬레이션을 제시한다. 또한 부품상태기반 시뮬레이션을 생산시스템의 일정계획에 적용하여 개발된 일정계획 시뮬레이터를 제안하고 그 실행과정에 대하여 설명한다.

2. 관련 연구 현황

시물레이션이란 전체 시간에 발생하는 사건에 대해 모델이 실제 시스템의 반응을 모방하도록 하여, 프로세스나 시스템을 모델링하는 것을 의미한다[1]. 실제 시스템에서 발생하는 개별 사건들에 대하여 정의하고 각 사건에 의하여 시물레이션이 진행되는 사건기반 시물레이션이 시물레이션 프로그램의 주된 흐름을 이루고 있다.

이와 같은 사건기반 시물레이션 프로그램에서 시물레이션을 수행하기 위하여 대상 시스템에 대한 모형을 보다 편리하게 작성하기 위한 많은 연구들이 진행되었다. Au는 사각형(rectangle), 원(circle)과 유방향선(directed arc)으로 구성되는 Activity Cycle Diagram을 이용하여 시물레이션 모형을 작성하는 방법에 대하여 연구하였다. Au는 임의의 시스템이든지 Entity, Activity와 Queue를 이용하여 시물레이션 모형을 작성하는 MacACD라는 프로그램을 개발하였다[2]. Kienbaum은 Activity Cycle Diagram의 세 가지 구성요소를 보다 세분화시켜서 Activity Cycle Diagram의 계층적 구조를 제시하였다. 이런 구조를 이용하여 시물레이션 모형을 Network형태로 표현하는 ADOO를 개발하여 실제로 시물레이션 모형을 작성하였다[3]. Hirata는 객체지향프로그래밍 관점에서 Activity Cycle Diagram을 이용하여 시물레이션 모형을 구축하는 방법에 대하여 연구하였다. 또한 그는 모형 작성의 결과물을 Pseudo code로 작성하였는데, 그가 작성한 모형은 기존의 상용 시물레이션 소프트웨어중의 하나인 MODSIM II에서 사용할 수 있는 모형으로 쉽게 변환되었다[4].

또한 객체 지향 관점으로 생산시스템에 대한 시물레이션 프로그램 개발에 접근한 연구가 있었다. 지능을 부여한 생산시스템의 시물레이션을 수행하기 위해, Agent 개념을 이용하여 개발된 Laval이 있었으며, 실시간으로 대화형 시물레이션을 구현하기 위한 OOSIM이 있다. 시물레이션 전문가가 아니라도 쉽게 시물레이션 모형을 구축할 수 있도록 아이콘을 이용하여 모형을 작성할 수 있도록 개발된 SmartSim등이 대표적인 예이다[5]. 이와 같이 최근에 개발되고 있는 시물레이션 프로그램은 해당 분야의 필요성에 따라서 각자의 영역에 맞게 개발하고 있다.

Narayanan은 최근에 개발된 대표적인 객체지향 생산시스템 시물레이션 프로그램으로 [표 1]과 같은 6가지의 시물레이션 프로그램을 선택하여 정리한 바 있다[6]. [표 1]의 생산시스템 시물레이션 프로그램은 동일하게 객체지향의 관점에서 설계되었지만 개발한 사람의 관점과 용도에 따라서 모두다 서로 다른 구조를 가지고 있다.

OOSIM은 생산시스템의 구성요소를 Plant와 Controller 클래스로 나누어 정의하고 생산제품 및 원자재에 대해서는 독립적인 클래스를 만들어 Material 클래스로 정의하였다[7, 8].

Smart/Sim에서는 생산시스템 시물레이션의 모든 구성요소를 Object 클래스로부터 상속받은 클래스로 보았다. 시물레이션을 실행하는 Simulator, 부품을 나타내는 Part, 사건기반 시물레이션을 위한 Event와 생산시스템의 구성요소를 나타내는 StationarySimulationObject클래스로 대별하고 있다. 또한 생산시스템의 구성요소는 Router, Source, Sink, Workstation, StorageFacility, Subsystem등의 클래스를 정의하였다[9].

BLOCS/M은 시물레이션 프로그램의 구성요소를 Resource와 FutureEvent클래스로 나누고, Resource로부터 BasicLot와 Workstation클래스를 상속받아 생산시스템을 모형화하였다[10, 11, 12]. 시물레이션 진행을 위해서는 FutureEvent클래스의 객체를 생성하여 이용하고 있다. 이와 같은 시물레이션 프로그램의 구조는 [그림 1]과 같다.

위에서 보인 객체지향 생산시스템 시물레이션 프로그램 외의 다른 구조를 가지는 시물레이션 프로그램도 있다. Iassinovski도 객체지향 시물레이터를 구축하였는데, 그는 Resource클래스로부터 부품과 재공품 등을 의미하는 TemporalResource와 기계를 의미하는

이름	개발 대학	Platform
BLOCS/M	University of California, Berkeley	Objective-C, ICPak201
DEVS	University of Arizona	SCOOPS
Laval	Laval University	Smalltalk-80
OOSIM	Georgia Institute of Technology	AT&T 2.1 C++
OSU-CIM	Oklahoma State University	Smalltalk-80
SmartSim	University of Michigan, Dearborn	Smalltalk-80

[표 1] 객체지향 시물레이션 프로그램

PermanentResource 클래스를 상속받은 체계를 이용하여 시물레이션 프로그램을 만들었다. Iassinovski는 시물레이션에서의 자원 개념과 객체지향관점에서의 객체 개념의 유사성에 주목하고, parameter와 attribute의 유사성을 강조하면서, 시물레이션과 객체지향의 개념의 동일한 점을 제시하였다. 그는 생산시스템 시물레이션이 수행중인 상태에서 기계의 상태를 FREE, BUSY, SETUP, PROCESS 상태로 분류하고, 각 이벤트에 따라서 기계의 상태가 변동되는 형식의 사건 기반 시물레이션의 구조를 따르는 시물레이션 프로그램을 개발하였다[13].

3. 객체의 상태와 사건

객체란 시스템을 구성하는 물리적인 구성요소이다. 그리고 이런 객체들은 대별하여 두 가지의 종류로 분류될 수 있다.

- 부품 객체
- 기계 객체

부품 객체는 생산시스템내에서 작업이 수행되어지는 객체이다. 원자재, 재공품, 부품들이 한 부품종류의 요소이고, 이들에 대하여 특정한 작업시간동안 작업을 수행하는 작업기계나 운반을 담당하는 운송수단 및 기계에서의 작업을 대기하는 대기행렬등이 기계종류를 이룬다.

일정계획의 정의가 "일련의 작업이 수행되도록 생산자원을 시간에 따라 할당하는 의사결정과정[14]"이듯이 생산시스템의 객체는 대표적으로 부품 객체와 기계 객체로 나눌 수 있다. 또한 Smith등은 Shop Floor는 부품 객체와 기계 객체의 상호작용의 집합으로 표현될 수 있다[15]고 언급한 바와 같이 생산시스템의 가장 중요한 객체는 기계 객체와 부품 객체이다.

객체는 특정한 시점에 하나의 상태를 가진다. 사건기반 시물레이션에서 하나의 사건에 의하여 부품과 기계의 상태가 변한다고 할 때 상태는 해당 상태에 주어지는 조건에 의하여 다른 상태로 전이하게 된다.

사건기반 시물레이션에서 사건은 사건이 발생할 시간과 사건이 발생할 시점의 직전 상황과 사건이 발생한 이후의 상황으로 이루어진다.

$$e = \langle t_e, C^-_e, C^+_e \rangle$$

t_e 시점에 발생하는 사건은 C^-_e 의 상황을 C^+_e 의 상황으로 만들게 된다.

하나의 사건은 하나의 시스템내 객체에 의하여 실행되어서는 측면이 강하다. 하나의 사건이 실행되면 사건을 실행하게된 객체의 상태가 변하면서 기타의 다른 시스템내 다른 객체들의 상태를 변화시킬 수도 있다. 사건에 의하여 시스템 및 객체의 상태가 변동되는 것을, 객체의 상태에 의하여 사건이 발생하는 구조로 이해할 수도 있다.

$$s = \langle s_c, t_s, \langle c_i, v_i, s_i \rangle \rangle$$

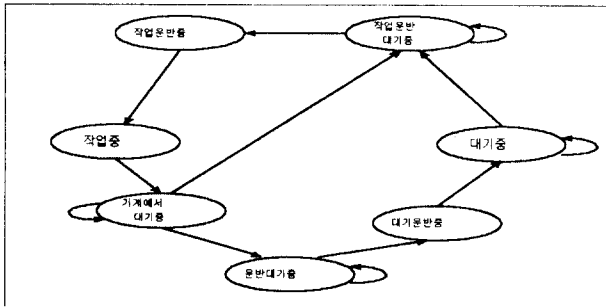
객체의 상태는 현재의 상태(s_c)와 현재 상태를 벗어날 시점(t_s)로 구성된다. 다음으로 현재 상태로 부터 벗어날 시점의 상황(c_i)와 이 상황에서 발생할 사건(v_i)에 의하여 다른 상태(s_i)로 전이할 수 있다.

하나의 상태로 부터 전이될 수 있는 상태는 현재 상태를 벗어날 시점의 상황에 의하여 결정된다. 그리고 이 시점에 적용되는 사건은 기존의 사건기반 시뮬레이션에서 적용될 수 있는 사건과 동일하다. 그러나 사건에 대하여 직접 접근하는 것보다 하나의 상태를 기준으로 접근함으로써 개별 사건들을 보다 쉽게 정의할 수 있게 된다.

4. 생산시스템 일정계획 시뮬레이터

4.1 객체 상태 전이

FMS의 경우 부품 상태의 전이를 개략적으로 보면 [그림 1]과 같다.



[그림 1] FMS에서의 부품객체 상태 전이도

- 작업중 : 부품이 기계에서 작업중이다.
- 대기중 : 부품이 AS/RS 혹은 다른 장소에서 작업을 받지 않고 대기중이다.
- 기계에서 대기중 : 부품이 특정한 기계에서 작업을 마치고 기계에서 운반수단을 기다린다.
- 작업운반대기중 : AS/RS에서 혹은 특정한 기계에서의 작업을 마치고 다음 기계로의 이동을 위한 운반수단을 할당받은 후 운반수단이 도착하기를 기다린다.
- 작업운반중 : AS/RS 혹은 특정한 기계에서 작업을 마치고 다음 기계에서의 작업을 위하여 운반중이다.
- 대기운반중 : 기계에서 작업을 마치고 AS/RS로 운반중이다.
- 운반대기중 : 기계에서 작업을 마치고, AS/RS로 이동하기 위해 운반수단을 할당받은 후 운반수단이 도착하기를 기다린다.

각 부품객체는 7가지의 상태 중에서 한 가지를 부품객체의 상태로 갖게되면, 그 부품객체의 다음 상태는 생산시스템의 상황에 따라서 각각 결정된다. 예를 들어 기계에서 '대기중'인 부품 객체는 다음에 작업할 기계 객체의 상태가 대기중이라면 작업운반대기중 상태로 변하겠지만, 대기중이 아니고 작업중 혹은 할당된 상태라면 대기운반중 상태로 바뀔 것이다. 이외의 다른 상태라면 다음 상태는 [그림 1]과 같이 한 가지 상태로 결정된다. 위와 같은

부품 상태에 대응하여 각 기계들은 작업중, 대기중, 할당된 상태중 한 가지 상태를 가지게 된다.

[표 2]는 특정한 상태의 부품객체가 전이시각에 도달했을 때, 각 부품객체가 처해진 상황을 요약하여 정리한 표이다. 이와 같이 처해진 상황에 대응하여 다음의 상태로 변하고, 각 변수들을 수정하게 된다.

부품 객체들이 한 가지 상태에서 다른 상태로 전이될 때에는 다양한 값들이 변하게 된다. 제일 먼저 부품 객체의 상태가 변하게 되고, 동시에 부품 객체의 잔여작업시간과 잔여작업수들이 변한다. 뿐만 아니라 생산시스템내의 다양한 상태변수값이 수정되어 전체 잔여작업수와 전체 잔여작업시간 등의 변수들이 변하게 된다. 이런 과정을 수행하는 기능들을 각 부품 객체와 기계 객체에 부여하여 일정계획 시뮬레이터를 구축하였다.

[표 2] 부품객체 i 의 상태별 전이시각 시점의 의미

부품객체의상태	상황
작업중	i 부품이 작업종료
대기중	i 부품의 다음 작업기계가 작업종료
기계에서 대기중	i 부품 운송 수단 할당 대기
작업운반중	i 부품이 작업할 기계에 도착
대기운반중	i 부품이 AS/RS에 도착
운반대기중	i 부품을 AS/RS로 운반할 운송수단이 도착
작업운반대기중	i 부품을 다음 작업기계로 운반할 운송수단이 도착

4.2 일정계획 시뮬레이터의 구성

본 논문에서 제시하는 시뮬레이터는 사건기반 시뮬레이션에서 이용하는 이벤트 리스트가 없다. 단지 부품 객체들의 상태 정보와 전이 시각을 가지고 있는 부품객체 리스트가 있을 뿐이다.

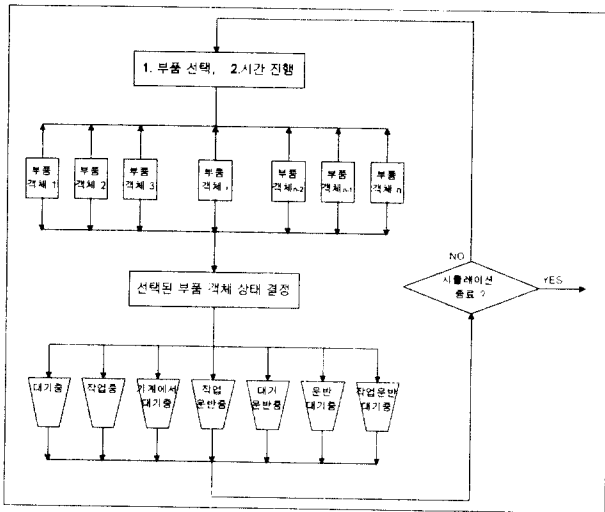
사건기반 시뮬레이션에서 다양한 사건들에 대하여 어떤 프로세스가 진행될 것인지 사전에 정의가 되어 있다면, 객체기반 시뮬레이션에서는 각 객체들의 상태에 의하여 사전에 정의된 순서대로 시간지연이 이루어지게 된다. 사건기반 시뮬레이션에서 하나의 이벤트가 실행되면 이와 연관된 각종의 데이터들이 수정된다. 이에 대하여 부품상태기반 시뮬레이션에서는 하나의 부품 객체의 상태에 따라서 관련있는 데이터들을 직접 수정하게 된다.

앞 절에서 설명한 바와 같이 모든 객체들은 각각 그 상태와 그 상태로부터의 전이시각을 가지고 있다. 모든 객체들 중에서 가장 빠른 전이 시각을 가지고 있는 부품객체에 의하여 시간 진행이 이루어진다. 부품 상태 기반 시뮬레이션은 [그림 2]와 같은 구조를 따르면서 다음과 같은 순서에 따라 시뮬레이션이 진행된다.

- Step 1 : 부품 객체들 중에서 전이 시각이 가장 빠른 객체를 선택하고 시뮬레이션 시각을 객체의 전이시각으로 한다.
- Step 2 : 부품 객체의 상태를 판단한다.
- Step 3 : 부품 객체의 상태에 따라서 각각 해당하는 루틴을 수행한다.
- Step 4 : Step 1로 간다.

선택된 부품의 전이 시각은 선택된 부품의 상태에서 다른 상태로 전이할 시각을 의미한다. [표 2]에서 부품객체가 전이시각에 직면한 상황을 설명하였다. 이 상황에서 각 상태별로 수행에 해당하는 루틴을 수행한 다음 그 결

과로서 부품 객체 변수와 기계객체 변수를 수정한다.



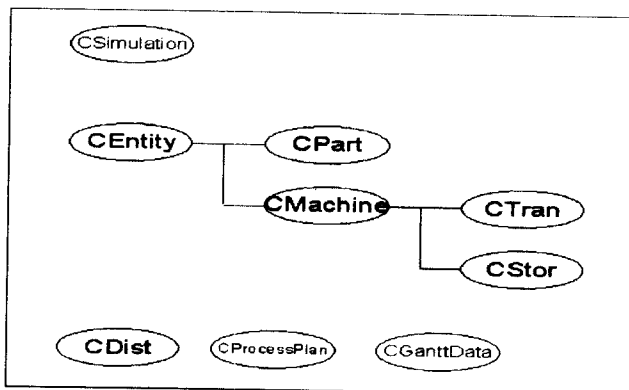
[그림 2] 부품객체상태 기반 시뮬레이터의 실행 구조

4.3 일정계획 시뮬레이터의 클래스 구조

본 논문에서의 일정계획 시뮬레이터는 Visual C++ 5.0에서 구축되어졌다. 객체지향 프로그램 개발 기법에 기반하여 작성되었으며, 생산시스템의 두 가지 객체 개념에 기반하여, 다음과 같은 클래스 구조를 설계하였다.

CEntity 클래스는 생산시스템내의 모든 객체에 대한 클래스이다. CEntity로부터 상속받은 기계객체 클래스로 CMachine 클래스가 있으며, 부품객체 클래스로 CPart 클래스가 있다. 또한 기계객체 클래스로부터 상속된 CTran 클래스와 CStor 클래스가 있다. CTran 클래스는 운송수단을 나타내고, CStor 클래스는 AS/RS를 나타낸다.

CSimulation 클래스는 시뮬레이션을 수행하기 위한 클래스이고, CDist는 난수발생과 확률변수 생성을 위한 클



[그림 3] 일정계획 시뮬레이터의 클래스

래스이다. 이에 더하여 CProcessPlan은 부품 객체가 생산시스템내에서 가공받을 순서를 저장하기 위한 클래스이고 CGanttData 클래스는 시뮬레이션 결과를 Gantt Chart로 그리기 위한 클래스이다.

CSimulation은 CPart클래스로 이루어진 부품 리스트와 CMachine클래스로 이루어진 기계 리스트를 멤버로 가진다. 또한 확률분포생성을 위한 CDist클래스의 객체를 멤버로 가진다. Gantt Chart생성을 위하여 CGanttData클래스로 이루어진 리스트를 멤버로 가진다.

CEntity 클래스의 멤버변수로는 부품객체의 상태를 나타내는 변수와 부품객체의 전이시각을 나타내는 변수를 기본으로 가지고 있으며, 각 객체의 Identifier를 동시에

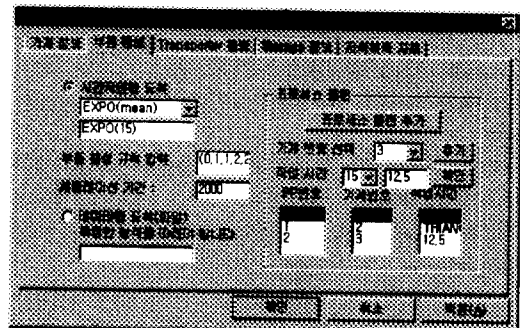
가지게 된다. 그리고 CPart클래스는 부품가공순서를 위한 CProcessPlan 클래스의 객체를 멤버로 가지게 된다.

4.4 일정계획 시뮬레이터 실행

제안한 일정계획 시뮬레이터는 생산시스템에 대한 자세한 모형화 작업을 필요로 하지 않는다. 본 논문에서의 모형화 방법은 기존의 ARENA와 AutoMod와 같은 범용 시뮬레이션 프로그램과 달리 시뮬레이션을 할 시스템의 구성을 선택하면서 시작한다. 생산 시스템을 크게 Job Shop, FMS Type, Flow Shop 모두 세 가지 시스템으로 분류하고 이러한 생산시스템을 그 대상으로 하였다. 이제 세 가지 형태의 생산 시스템 중에서 한 가지를 선택하면 그에 해당하는 자료 입력 다이얼로그가 나타나고 이 다이얼로그에 대상으로 하는 생산시스템에 대한 정보를 입력한다.

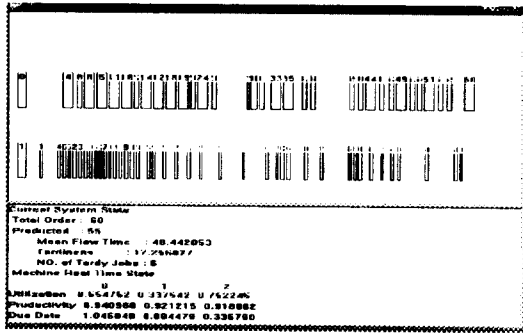
자원이라고 할 수 있는 기계에 대한 자료로서 기계 대수와 각 기계로 부품이 투입되는 방법과 각 기계에서 작업이 완료된 부품이 기계에서 나오는 방법을 입력한다. 각 기계간의 부품 운반방법에 대한 입력이다. 작업이라고 할 수 있는 부품에 대한 정보는 2가지 형태의 부품 투입에 대하여 자료를 입력할 수 있다. 일정한 간격으로 부품이 투입되는 형태와 시뮬레이션 초기에 모든 부품이 투입되는 형태, 2가지 형태의 부품 투입에 대하여 모델링된다. 전자의 형태로 부품이 투입되는 경우에는 각 부품별 Part Mix와 각 부품별 프로세스 플랜을 입력한다. 후자의 형태로 부품이 투입되는 경우에는 각 부품별 프로세스 플랜을 파일 형식으로 입력하도록 한다.

[그림 4]는 사용자가 평균이 15인 지수분포간격으로 생산시스템안으로 부품이 투입되고, 3가지의 부품을 1:2:1의 비율로 생산하며, 시뮬레이션 실행 기간은 2000 시간 단위이고, 0번의 프로세스 플랜을 가지는 부품은 0, 2, 3번 기계에서 각각 Triangular분포와 12.5 시간 단위의 작업을 한다는 사항을 입력한 것이다.



[그림 4] 부품데이터 입력 1

[그림 5]와 같은 화면이 나타나면서 시뮬레이션이 수행된다. 시뮬레이션이 수행되는 내역이 Gantt Chart에 의해 나타나고 동시에 생산시스템의 실제 상황이 화면에 나타난다. 시뮬레이션이 수행되는 현 시점에서 볼 때, 모두 60개의 부품이 생산시스템으로 투입되었으며, 55개의 부품이 생산 완료되었다. 생산된 부품의 평균체류시간은 40.442053이고, 평균 Tardiness는 17.256877이며, 현재까지 6개의 납기지연된 부품이 발생했다. 그리고 각 기계별 Utilization 및 기타 수행도를 화면에 실시간으로 보여주고 있다.



[그림 5] 시뮬레이션 수행화면

5. 결론

본 논문에서는 시뮬레이션 프로그램 개발에 있어서 사건을 정의하고 사건 종류에 의하여 시뮬레이션을 진행하는 접근 방법을 대신하여, 시스템내의 객체상태에 기반하여 시뮬레이션 프로그램을 개발하는 방법을 제시하였다. 또한 제시하는 방법에 의하여 생산시스템의 일정계획 시뮬레이터를 개발하였다.

6. 참고 문헌

- [1] Pegden, C. Dennis, Robert E. Shannon and Randall P. Sadowski, "Introduction to Simulation Using SIMAN", 2nd Ed., McGraw-Hill, 1995
- [2] Au, Grace and Ray J. Paul, "Graphical Simulation Model Specification Based on Activity Cycle Diagrams," Computers & Industrial Engineering, VOL.26, NO.2, pp.295-306, 1994
- [3] Kienbaum, Germano and Ray J. Paul, "H-ACDNET: An Object-Oriented Graphical User Interface for Simulation Modelling of Manufacturing Systems," Simulation Practice and Theory, VOL.2, pp.141-157, 1994
- [4] Hirata, Celso M. and Ray J. Paul, "Object-Oriented Programming Architecture for Simulation Modelling," International Journal in Computer Simulation, Vol 6, pp. 269-287, 1996
- [5] Ulgen, Onur M., Timothy Thomasma and Youyi Mao, "Object Oriented Toolkits for Simulation Program Generators," Proc. of the 1989 Winter Simulation Conference, Edit(E.A. MacNair, K.J. Musselman, P. Heidelberger) pp. 593-600, 1989
- [6] Narayanan, S., D.A. Bodner, U. Sreekanth, T. Govindaraj, L.F. McGinnis and C.M. Mitchell, "Research in Object-Oriented Manufacturing Simulations: An Assessment of the State of the Art," IIE Transactions, VOL.29, NO.7, pp.507-526, 1997
- [7] Narayanan, S., D.A. Bodner, U. Sreekanth, T. Govindaraj, L. F. McGinnis and C. M. Mitchell, "Modelling Control Decisions in Manufacturing Systems Simulation Using Objects," Proc. of the 1994 IEEE International Conference on Systems, Man, and Cybernetics, San Antonio, TX, October 1994, pp.1392-1397, 1994
- [8] Bodner, Douglas A., T. Govindaraj, Leon F. McGinnis and Christine M. Mitchell, "An Integrated Approach to Modeling Distributed Manufacturing Control and Human Operators," Proc. of the 1995 IEEE International Conference on Systems, Man, and Cybernetics, Vancouver, BC, Oct.22-25, 1995, pp.3437-3442, 1995
- [9] Ulgen, Onur M., and Timothy Thomasma, "SmartSim: An Object Oriented Simulation Program Generator for Manufacturing Systems," International Journal of Production Researches, VOL.28, NO.9, pp.1713-1730, 1990
- [10] Adiga, Sadashiv, "Software Modelling of Manufacturing Systems: A Case for an Object-Oriented Programming Approach," Annals of Operations Research, VOL.17, pp.363-373, 1989
- [11] Adiga, Sadashiv and C. Roger Glassey, "Object-Oriented Simulation to Support Research in Manufacturing systems," International Journal of Production Researches, VOL.29, NO.1, pp.2529-2542, 1991
- [12] Glassey, C.R. and S. Adiga, "Berkeley Library of Objects for Control and Simulation of Manufacturing(BLOCS/M)," Applications of Object-Oriented Programming(Editor : Pinson, L.J. and R.S. Wiener), Addison Wesley, pp.1-27, 1990
- [13] Iassinovski, S.I. and V.V. Emelyanov, "An AI-based object-oriented tool for discrete Manufacturing system simulation," Journal of Intelligent Manufacturing, VOL. 8, pp. 49-58, 1997
- [14] 대한산업공학회(편), 산업공학용어사전, 청문각, 1993.
- [15] Smith, Jeffrey S. and Sanjay B. Joshi, "Objected Oriented Development of Shop Floor Control Systems for Computer Integrated Manufacturing," Proc. of International Conference on Object-Oriented Manufacturing System, Calgary, Canada, May 3-6, pp.152-157, 1992