

대량생산 체제의 Simulation을 위한 Activity중심 Modeling (Activity-oriented Modeling of Mass Production System)

최병규*
박성주*
신하용*

Abstract

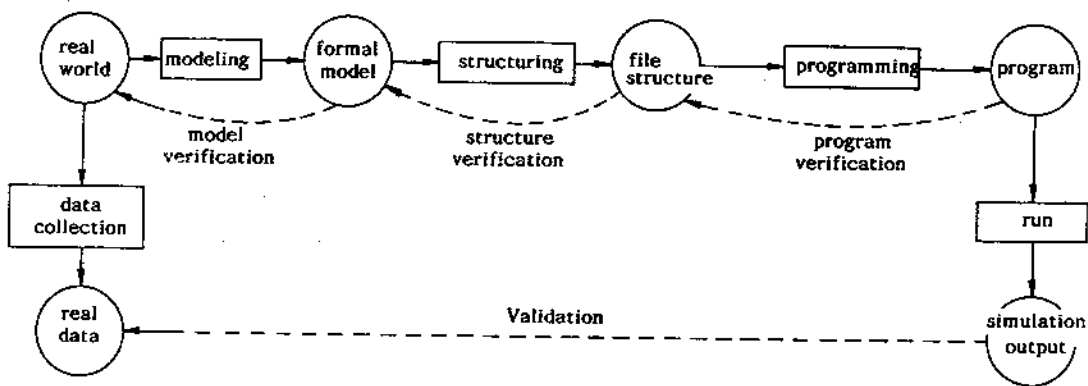
Described in this paper is a modelling methodology for mass production system simulation. The mass production system under consideration consists of various types of flow lines, special purpose production facilities, conveyor lines, palletized carts, and storage facilities. This type of production system is typical in home appliance industry, automobile industry, footwear industry, etc. where a variety of product mix are mass-produced.

The modelling methodology is based on the "discrete-event formalism", and an "activity-oriented world view" is adopted to formalize the system description. A distinctive feature of the modelling methodology is that only the static structure (ie, system components) is included in the fixed model. The dynamic structure of the system is specified through a "data-driven" mechanism, which is an extension of the "experimental frame" concept. Each type of system components (ie, flow line, conveyors, carts, etc.) is formally modeled by using Activity Cycle Diagrams. The issue of "model structuring" is also addressed. The modeling methodology has been successfully applied in a real simulation study of a mass production system.

소비자들의 다양한 선호를 만족시키면서 날로 격심해지는 가격경쟁에서 우위를 유지하기 위하여는 "유연성 있는 대량생산체제"의 도입이 요구된다. 전통적으로 대량생산체제는 Flow shop의 형태를 갖어왔는데, 주어진 생산체제에서 다양한 제품모델들을 동시에 생산

할 필요가 증대됨에 따라 기계부품의 가공을 위하여는 FMS(flexible manufacturing system)의 개념이 널리 도입되고 있으며, 최종제품의 생산을 위하여는 JIT(just-in-time)생산체제의 개념이 크게 각광을 받고 있다. 이러한 새로운 형태의 생산체제가 추구하는 바는 여러

* Korea Advanced Institute of Science and Technology



〈그림 1〉 Modeling과 Simulation의 Cycle

종류의 제품모델들을 적기에 생산하면서 재공 재고를 최소화 하는 것이다.

유연성 있는 대량 생산체제는 막대한 설비투자가 소요되고 매우 복잡한 일정계획의 수립이 요구된다. 따라서 생산체제의 설계분석과 운용의 효율화를 위한 도구가 필요한데, 문제의 복잡성 때문에 이러한 분석을 위한 도구로는 컴퓨터 시뮬레이션이 적합하다[2].

FMS의 설계 분석을 위하여는 컴퓨터 시뮬레이션이 널리 적용되고 있으며 [4], 상당수의 시뮬레이션 패키지들이 개발되었다[1]. 그러나 최종제품 조립을 위한 대량 생산 체제에 관한 시뮬레이션 연구는 부진한 편이며 최근에야 JIT 생산체제에 관한 시뮬레이션 사례가 소개되고 있다[18].

본 논문에서 다루고자 하는 대량생산체제는 여러 형태의 flow line, 특수 생산설비(기계), conveyor line, 팔렛용 운반대차, 저장설비 등으로 구성된 최종제품의 생산을 위한 공장을 의미한다. 이러한 대량생산 체제는 부분적으로는 상당한 자동화가 이루어져 있고, 여러 종류의 제품모델들을 생산할 수 있도록 구성되어 있다. 우리나라에도 이러한 형태의 공장이 큰 비중을 차지하고 있는데 자동차, 자전거, 냉장고 및 TV 등의 가전제품, 신발류, 의류(기성복)등을 생산하는 공장들이 이에 해당된다.

본 연구는 시스템 모델링의 측면에서 일반적으로 대량생산체제가 갖는 특성을 규명하고, 나아가 이에 적합한 모델링 방법 및 시뮬레이션 프로그램 개발 방안을 제시해 보기 위한 것이다. 이를 위하여 일반적인 시스템 이론에서 출발하여 체계적인 모델링 방법을 찾아보도록 노력하였다. 본 논문은 크게 세 부분으로 구성되어 있는데 첫 번째 부분에서는 대량생산 체제의 모델링을 위한 접근방법과 모델 기술 방법을 다루고 있으며 두 번째 부분에서는 대량생산 체제를 구성하고 있는 component들에 대한 modeling의 방법론을 제시하고 있다. 세 번째 부분에서는 대량생산을 위한 공장 전체적인 modeling을 구체적인 예를 들어 보이고, 또 그것을 시뮬레이션하기 위한 구성방향을 기술하고 있다.

1. 모델링 및 시뮬레이션 흐름도

분석의 대상이 되는 실제문제(real world)에 대한 시뮬레이션 프로그램을 작성하기 위하여는 system modeling, model structuring, programming의 세 단계를 거치는 것이 바람직하다. 이렇게 여러 단계로 분리하여 접근하는 이유는 모델을 구하고 시뮬레이션 프로그램을 개발하는 절차를 체계화하고 나아가 각 단

계에서 얻어진 모델(formal model, structured model, program model)에 대한 검토를 용이하게 하기 위함이다.

(그림1)에는 모델링 및 시뮬레이션 부분을 위한 전반적인 흐름도가 나타나 있다. 그림에서 “modeling”이란 실제의 대상 시스템을 분석의 목적에 적합한 하나의 개념적 모델로 파악하여 이를 일정한 표시 방식에 따라 체계적으로 기술하는 것을 의미하며, “structuring”이란 얻어진 “formal model”을 프로그래밍이 용이하도록 소프트웨어적인 측면에서 재구성하는 것을 의미한다. 시스템의 모델링 단계에서 고려되어야 할 사항들은

- ① modeling formalism [20, p.24]의 선정
- ② formal model에 대한 system boundary의 결정[9]

③ 적합한 “World view” [23, 24]의 선정 등이다. 대량생산 체제는 그 성격상 이산체제(discrete-event system)이므로 modeling formalism으로는 discrete-event model이 적합하다. System boundary의 결정과 world view의 선정은 분석 대상의 성격에 따라 여러 가지 가능성이 있으므로 이들에 대하여는 다음 절 등에서 보다 상세히 언급하고자 한다.

2. 대량생산체제의 System Boundary 결정

대량생산체제의 각 구성요소(생산설비, 운반설비, 저장설비, 공장 배치, routing, 일정 계획 등)를 formal model structure의 각 요소에 mapping시키는 것을 system boundary 결정이라 한다. 이산체제는 finite state machine [5, p.15]과 유사한 구조를 가지며 다음과 같이 정의된다. [20, p.28, 24].

$M = \langle X, S, \delta, \tau \rangle$: discrete event system (1)

X : external event set

S : sequential state set

δ : transition specifying function

τ : time advance function

$\Omega = \{w \mid w : \langle t_0, t_1 \rangle \rightarrow X\}$
: input segment set

$Q = \{(s, e) \mid s \in S, 0 < \tau(s)\}$
: system state set

$\delta : Q \times X \rightarrow S$: system state transition

$\lambda : Q \rightarrow Y$: output generation function

위의 정의에 따르면 Discrete-event system M 은 입력 Ω 에 대하여 출력 Y 를 실현시키는 일종의 finite state machine으로 간주될 수 있다. 한편 하나의 시스템을 구성하는 개별 구성요소들을 규정짓는 부분을 모델의 “static structure”라고 하며 구성요소들 간의 상호관련도를 규정하는 rule들을 “dynamic structure”라고 하는데 [10], 식(1)의 경우에는 X, S 가 static structure에 해당되고 δ, Ω, τ 는 dynamic structure에 해당된다. 좁은 의미에서의 system boundary 결정은 생산체제의 구성요소들 중에서 sequential state set S 에 해당되는 부분을 결정하는 것을 뜻한다.

본 논문에서 고려하고 있는 대량생산체제의 경우에는 시뮬레이션 분석의 목적으로 설비용량결정, 공장배치결정, product mix에 따른 재고상태 파악, 일부 부품의 외주가공 검토 등이 포함되므로 system boundary(즉 static structure)는 생산설비, 운반설비, 저장설비의 집합으로 정의하는 것이 바람직하다. 한편 여러 품목을 생산하는 대량생산 체제의 경우에는 효율적인 일정계획의 수립이 매우 중요하므로 주어진 생산계획(혹은 Master production schedule)으로부터 상세한 일정계획을 수립하는 과정은 시뮬레이션을 별도로 수행하여 본 모델에 입력시키는 것이 바람직하다.

이상의 고찰을 토대로 discrete-event system의 각 요소들을 대량생산체제의 구성요소들과 연결시키면 다음과 같다.

$S = \{\text{생산설비, 운반설비, 저장설비의 상태}\}$

의 집합

$X = \{\text{생산가능한 제품들의 작업지시 집합}\}$

$\Omega = \{\text{일정계획의 집합}\}$

$\delta = (\text{공장 layout; 부품들의 flow route})$

한편 τ (time advance function)은 널리 알려진 event scheduling mechanism [7]을 사용하면 되는데 보다 상세한 사항은 참고문헌 [7, 24]을 참조하기 바란다.

3. 대량생산체제 모델링을 위한 World View의 선정

실제 시스템에 대한 논리적 모델을 세우기 위하여는 시스템의 구조를 어떤 관점에서 파악하느냐가 매우 중요하다. 이러한 관점의 선정을 보통 "World view"라고 한다 [23]. 지금까지 널리 사용되고 있는 World view에는 Event 중심, Activity 중심, Process 중심의 3가지가 있다[9]. 시뮬레이션 모델링 분야에 자주 쓰이는 용어들이 저자에 따라 서로 다른 의미를 갖는 경우가 많기 때문에 본 논문에서 사용되는 의미를 간략하게 정의하고자 한다.

Resource : 작업(혹은 서비스)을 수행하는 주체

Entity : 시스템내에서 서비스를 받는 대상

Station : Entity가 저장되거나 대기하고 있는 장소

State : Resource나 station의 현재 상태

Activity : Resource가 작업을 수행하는 행위

Event : 시스템의 state에 변동이 일어나는 것

Process : Entity가 일련의 service를 받는 과정

Event 중심의 World view는 "시스템 State의 변화"에 초점을 맞추어 모델링하는 것을 말하는데, 하나의 Event들을 "Event-Diagram" [12, 19]을 이용하여 표시하고 각 경우의 Event가 일어날 수 있는 조건들은 별도의 "Condition Table"에 나타낸다. Process 중심의 World view에서는 시스템내에서 서비

스를 받는 Entity의 흐름에 초점을 맞추어 모델링을 수행하는데, 얻어진 모델은 "Block Diagram" [12, 17]이나 "Network Diagram" [15, 16]으로 표시된다. 한편 Activity 중심의 모델링에서는 시스템내의 Resource들이 서비스를 수행하는 행위에 관점을 두고 아울러 Entity의 흐름도 나타낸다.

World view의 선정기준은 앞서의 <그림1>에 나타난 바와 같이 "modeling", "structuring", "programming", "verification"등을 용이하게 할 수 있느냐에 있다. 앞서 언급된 바와 같이 대량생산체제에서는 Resource (생산설비, 운반설비, 저장설비)만을 모델의 static structure에 포함시키는 것이 바람직하므로 Process 중심의 World view는 부적합하다. Event 중심의 World view에서는 시스템의 state가 완전히 정의된 이후에 system state의 변동 과정을 기술하는데 역점을 두고 있으므로 modeling을 위하여는 부적합하다. 그러나 Event-Diagram과 Condition Table이 얻어진 이후에 실제 시뮬레이션 프로그램 작성하기에는 편리하다.

Activity 중심의 World view는 대량생산체제의 모델링에 가장 적합할 뿐 아니라 얻어진 모델의 검증(verification)을 용이하게 한다. 일단 World view가 선정되면 주어진 실제 시스템에 대한 모델을 체계적으로 기술할 도구가 마련되어야 한다.

4. Activity 중심의 모델 표현을 위한 도구의 개발

Activity 중심의 모델을 도식적으로 표현하기 위한 노력이 그동안 여러 사람들에 의하여 행하여져 왔는데 저자마다 상이한 이름을 사용하고 있지만 Activity Cycle Diagram (ACD)이라는 이름이 널리 쓰인다. ACD의 기본적인 형태 [6, 8]에서는 "idle state" (혹은 queue)를 원으로 표시하고, "active state"를 직사

각형으로 나타내, “state change”는 화살표를 사용하여 나타낸다. 그리고 어떤 Resource 혹은 Entity가 현재 위치해 있는 상태를 “marker”를 이용하여 표시한다. ACD에서의 “idle state”, “active state”, “marker”는 Petri-net [14]에서의 “place”, “transition” 및 “token”에 각각 해당된다.

ACD가 bipartite directed multigraph [14, p.11]의 형태를 유지하면 Petri-net 이론을 이용한 모델 검증을 체계적으로 수행할 수 있으며 ACD를 그대로 프로그래밍 언어로 번역하는 것도 가능하다[6]. 그러나 실제 시스템의 여러가지 상황을 충실히 모델로 표현하려면 bipartite graph로만은 표현되지 않는다. Torn [21, 22]은 기존의 Petri-net을 확장시켜서 “simulation net”이라고 불리고 이를 토대로 하여 automatic program generator의 개발을 시도하고 있으나 이 분야에 대한 연구가 요망된다.

본 논문에서는 기존의 ACD에 몇 가지 도식방법을 추가하여 사용하기로 한다. 대량생산체제의 모델에 적합한 Activity Cycle Diagram (ACD)은 다음의 사항들을 구분하여 나타낼 수 있는 것이 바람직 하다.

① Active state (Resource and Entity) : 서비스 수행중

② Non-active state (Resource) : Resource가 놓고 있거나 고장

③ State change (Resource) : Active state와 non-active state 간 변동

④ Entity flow (Unconditional) : 무조건 흘러가는 경로

⑤ Entity flow (Conditional) : 어떤 조건을 만족하는 Entity만 흘러가는 경로

⑥ Queue (Entity) : Entity가 일시 저장된 경우

⑦ Informatica (생산계획등)

⑧ Infinite capacity source (Entity) 이들에 대한 기호는 <표1>에 나타나 있다.

<표 1> Activity Cycle Diagram에 사용되는 기호

시스템 표현 내용	사용되는 기호
Active State	
Non-active State	
State Change	
Entity flow(Always)	
Entity flow(Conditional)	
Queue	
Information	
Infinite Capacity Source	

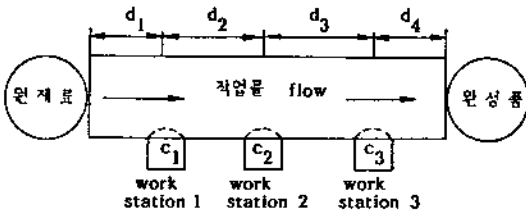
5. 대량생산체제를 구성하는 개별 설비의 모델링

대량생산체제의 모델링에는 Activity 중심의 World view가 적합하며 formal model을 기술하는 도구로는 ACD (Activity Cycle Diagram)을 사용할 수 있다는 점에 대하여 살펴 보았다. 본 절에서는 실제로 대량생산체제를 구성하고 있는 개별 설비들을 ACD로 표현하고자 한다. 일반적으로 대량생산체제는 flow line, 단독 기계, 저장 장소, 팔렛용 대차, belt conveyor, trolley conveyor 등으로 구성되어 있으며 이들 개별 설비는 Discrete-event System의 Static structure에 해당된다.

1) Flow line의 모델링

Flow line에서는 다수의 작업 station들이 하나의 이송체제에 순차적으로 연결되어 있으며 각 작업 station 사이에는 일정한 용량의 Buffer storage가 마련되어 있는 것이 보통이다. <그림2>에는 3개로 작업 station으로 구성된 flow line의 예가 표시되어 있다. Flow line은 이송체제 (transfer mechanism)의 형태에 따라 Continuous Flow Line (CFL),

Synchronized Transfer Line(STL), Free Transfer Line(FTL) 등으로 구별된다 [3].



〈그림 2〉 3개의 Work station을 갖는 Flow Line

CFL에서는 powered belt conveyor나 trolley conveyor에 의하여 부품이 이동되는데 신발류 제조 라인, 소형 기계부품 조립 라인, 대형 가전제품 생산라인, 자동차 내장품 조립 라인 등이 이에 해당된다. STL은 Indexing mechanism에 의하여 작동되는데 박판가공용 Press 라인, Progressive die 작업 라인, 자동차 차체용접 라인 등이 이에 해당되며 작업 station 간에 buffer storage가 없는 것이 일반적이다. FTL은 동력이 없는 roller conveyor 등에 의하여 수동으로 제품이 이동되는 Flow line을 의미하는데 TV 등 가전제품 조립라인, 수동 용접 라인 등이 이에 해당되며 보통은 buffer storage가 필요하다.

Flow line은 작업 station의 수, 각 작업 station의 cycle time, 작업 station 간의 이동 시간, 각 작업 station 사이의 buffer storage의 크기 등에 의하여 규정된다. 이러한 관계를 수식적으로 표현하면 다음의 식(2)와 같다.

$$\text{FLOWLINE} = (n, C, B, T) \dots\dots\dots (2)$$

n = no. of, work stations

$C = (c_1, c_2, \dots, c_n)$

$B = (b_1, b_2, \dots, b_n)$

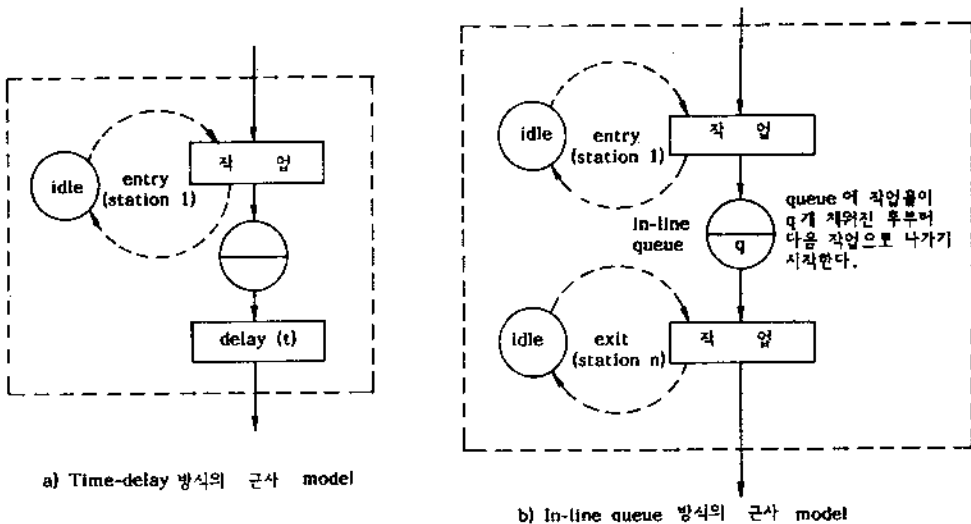
$T = (t_1, t_2, \dots, t_n)$

c_i = cycle time at station i

b_i = buffer storage for station i

t_i = transfer time from $(i-1)$ to i

본 논문에서 다루고 있는 대량생산체계의 경우에는 일반적으로 개별 flow line마다 수십 개의 작업 station을 갖는 경우가 보통이고 하나의 생산체제내에는 여러개의 flow line들이 있으므로 시뮬레이션을 효율적으로 수행하기 위하여는 개별 flow line을 근사적으로 모델링할 필요가 있다. Flow line에 대한 근사적 모델을 세우는 방식에는 두가지가 있을 수 있는데 하나는 부품이 flow line을 통과하는데

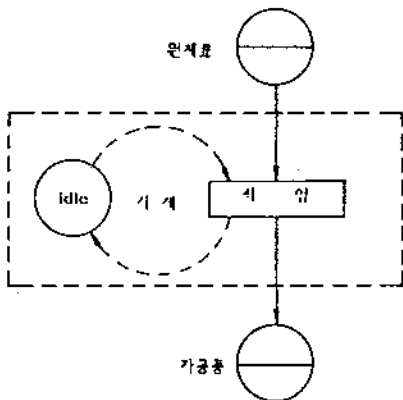


〈그림 3〉 Flow line의 근사모델 ACD (Activity Cycle Diagram)

걸리는 시간에 기준을 두는 방식이고, 다른 하나는 flow line 내에 머물 수 있는 작업물의 갯수에 기준을 두는 방식이다. 본 논문에서는 전자의 방식을 "Time-delay 방식"이라 부르고 후자를 "In-line-queue 방식"이라 부르고 있다.

〈그림 3〉에는 flow line에 대한 Time-delay 방식 근사모델과 In-line-queue 방식 근사모델을 보여주고 있다. Time-delay 근사모델은 flow line의 첫번째 작업 station에서 일정한 cycle time으로 작업물을 올려놓으면 얼마간의 지연시간 후에 작업물이 line을 빠져나가는 것으로 간주한다. 이 방식은 논리가 간단하고 실제 프로그램으로 옮기기 용이한 반면 전체 flow line 내의 작업물의 수가 많아지면 시뮬레이션 프로그램의 "event-list"가 매우 길어지고 따라서 계산시간도 길어진다. 단점이 있다. 따라서 Time-delay 방식의 근사모델은 line의 길이가 짧은 CFL과 STL을 표현하는데 적합하다.

In-line-queue 방식은 line의 길이가 긴 STL에 가장 적합하며 실제 프로그램으로 옮기기에 다소 복잡하지만 event-list를 짧게 해 주고 계산시간도 매우 짧다는 장점이 있다. 본 논문에서 제시한 근사모델은 일반적으로 flow line의 작업 station들의 cycle time에 변동



〈그림 4〉 단독 기계를 나타내는 ACD

이 심하고 buffer storage가 커지면 단순화에 따른 오차가 커진다.

2) 단독 기계의 모델링

단독 기계에 대한 ACD는 〈그림 4〉에 보인 바와 같이 매우 간단하다.

3) 저장 설비의 모델링

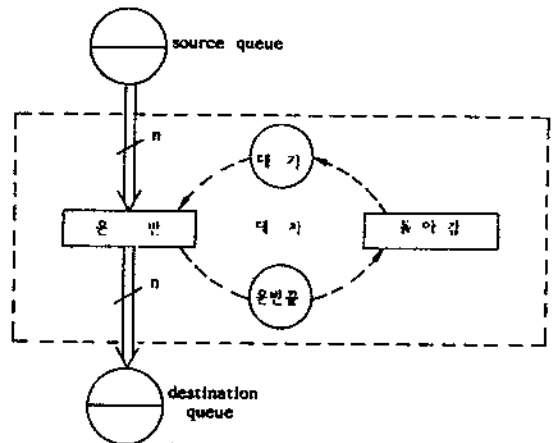
저장 설비는 단순히 하나의 Queue로 표시된다.

4) 펠릿용 운반대차의 모델링

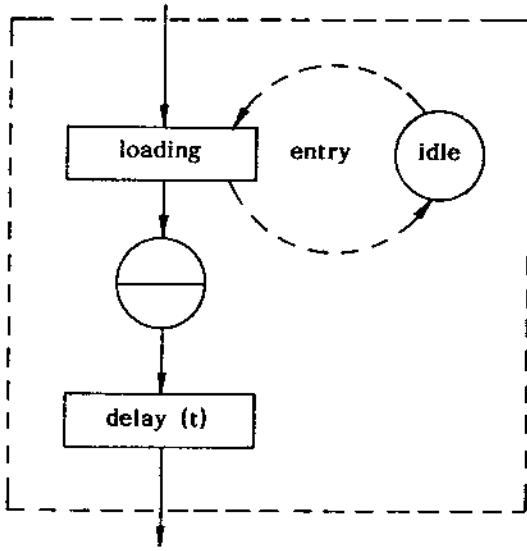
〈그림 5〉에 표시된 바와 같이 하나의 운반대차는 "운반", "운반끝", "돌아감", "대기"의 4가지 상태를 갖는다. 다만 한가지 특기할 사항은 팔렛(pallet)에 n개의 부품이 쌓이면 운반을 시작한다는 점이다. 〈그림 5〉는 source queue와 운반 사이가 n개의 arc로 연결되어 있어서 source queue에 part가 n개가 쌓이면 운반이 된다는 뜻이다.

5) 운반용 Trolley conveyor 및 Belt conveyor의 모델링

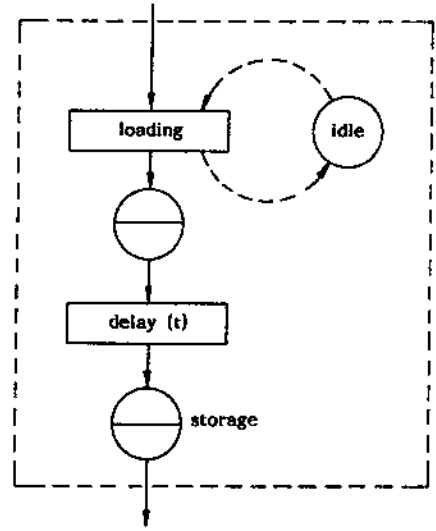
운반용 Trolley conveyor에 대한 ACD가 〈그림 6〉에 표시되어 있는데 이는 flow line에 대한 Time-delay 방식 모델과 동일하다.



〈그림 5〉 대차 운반의 ACD



〈그림 6〉 운반용 trolley 또는 belt conveyor의 ACD



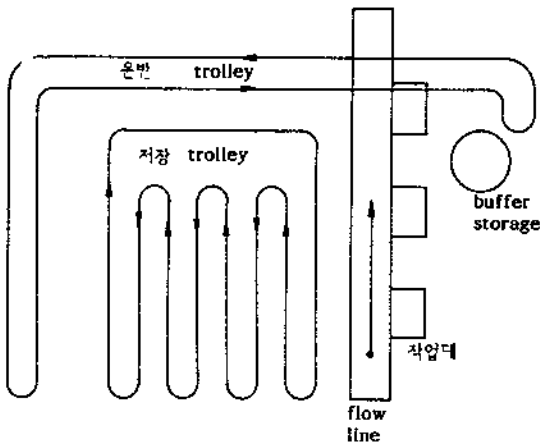
〈그림 8〉 〈그림 7〉에 대한 ACD

6) 운반 및 저장용 Trolley Conveyor

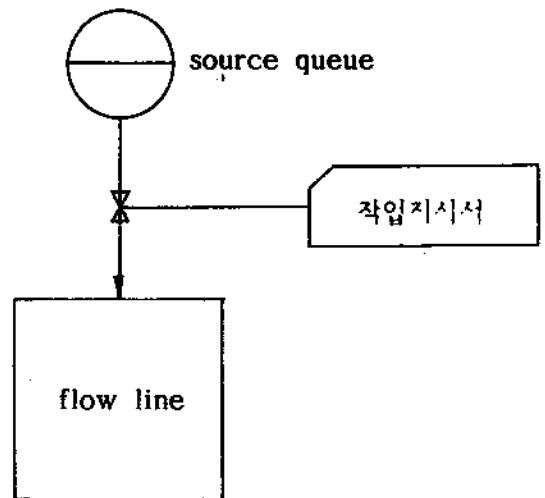
대량생산 체제의 부품의 부피가 크고 달리 쌓아두기 곤란한 경우에는 운반과 저장을 겸하는 trolley conveyor 체제가 널리 쓰인다. 〈그림 7〉에 보인 바와 같이 실제로는 저장 Trolley, 운반 Trolley, Buffer storage 가 하나의 시스템을 이루고 있는데 이에 대한 ACD는 〈그림 8〉과 같다.

6. 전체 시스템의 modeling 및 Structuring

앞 절에서 대량생산 체제를 구성하는 전형적인 요소들과 그것의 모델링에 대해서 살펴보았다. 본 절에서는 그러한 요소들간의 연결 및 전체 시스템의 모델링과 program 작성을 위



〈그림 7〉 운반 및 저장 trolley 시스템



〈그림 9〉 작업지시서 정보의 modeling

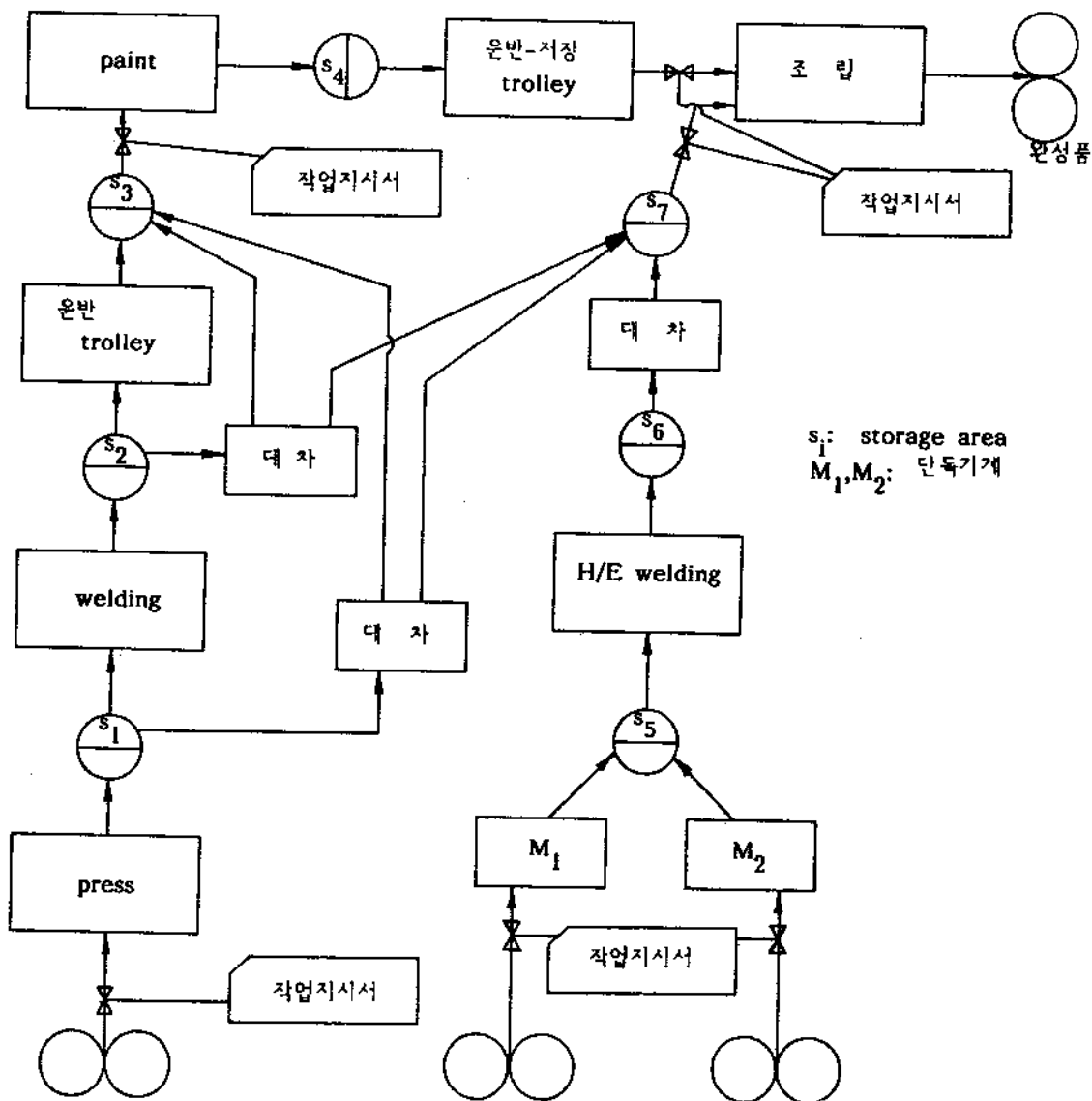
한 file structure에 관하여 기술하고자 한다.

대량생산체제의 경우, 생산 계획에 따른 작업지시가 line 단위로 내려가게 되므로 이러한 작업지시를 포함하여 모델링하는 것이 필요하다. <그림 9>는 source queue에 여러 종류의 part들이 섞여 있을 때, 작업 지시서에 따라 필요한 part들은 골라 flow line으로 보내는 작업을 모델링한 것이다.

공장 전체의 modeling은 각 구성요소의

<표 2> D 회사의 line 분류 및 모델

line 이름	갯수	line 종류	model
press line	3	STL	<그림3>, b)
용접 line	3	FTL	<그림3>, b)
열교환기용접 line	1	CFL	<그림3>, a)
paint line	1	CFL	<그림3>, a)
조립 line	3	CFL	<그림3>, b)
운반 trolley	1	CFL	<그림 6>



s_i : storage area
 M_1, M_2 : 단독기계

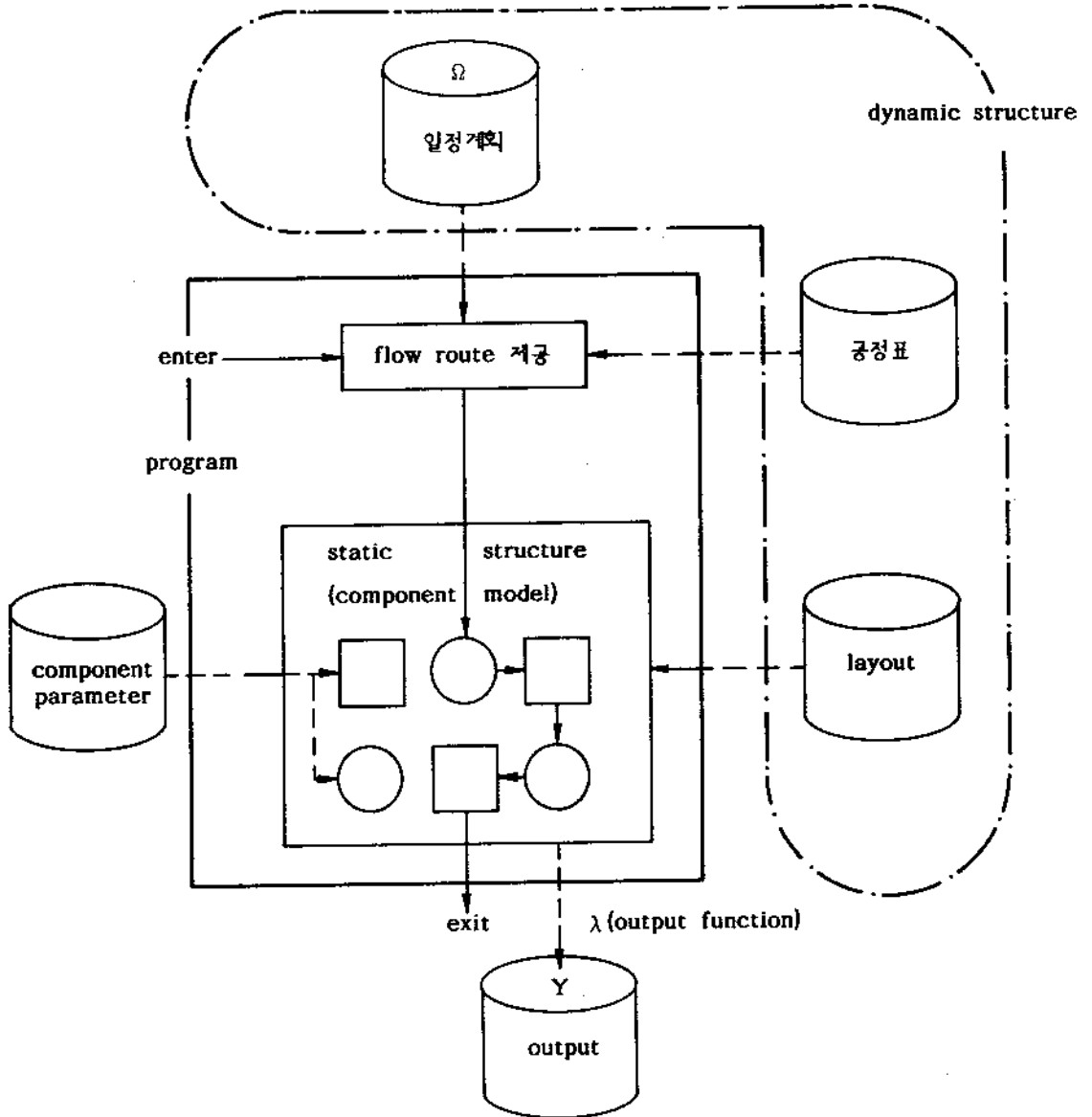
<그림 10> D사 가전제품 공장의 model

model을 연결해 놓으면 된다. 이것은 전체 시스템의 modeling을 단계적으로 수행할 수 있다는 면에서 매우 바람직한 성질이다. 공장 전체의 modeling의 예로 D회사의 한 가전제품 공장의 경우를 예로 들면 다음과 같다.

이 공장은 크게 4종류의 제품을 만들고 있으며, 한 제품은 5~15가지의 부품으로 이루어진

다. 생산 설비들은 <표 2>에 요약되어 있는 flow line들과 trolley 그리고 몇개의 운반 대차와 1개의 운반 및 저장 trolley system이 있다. 그밖에 몇 가지의 단독 기계가 있으나, model구성상 중요한 것만 포함시켰다.

<그림 10>은 이 공장의 설비들과 연관 관계를 modeling한 그림이다. <그림 10>에서



(그림 11) 전체 program의 file structure

press-S, welding line 과 조립 line 은 실제로는 각각 3개씩 있으나 그림이 너무 복잡하여 각각 하나씩으로 단순화시킨 것이다.

앞의 예와 같이 모델링된 전체 공장에 대한 simulation program 을 작성하기 위한 구조는 앞의 2절에서 설명한 static structure(X, S)와 dynamic structure(Ω, δ, τ)로 나누어 생각해 볼 수 있다. 이 system의 entity인 작업물은 일정계획(Ω)에 따라 system에 들어와서, 공정표에 있는 순서에 맞추어 layout을 따라 공장내의 설비들(S)을 거쳐 흘러 완성품이 되어 system을 빠져나간다. 이 중 ① 작업물의 흐름을 결정해서 component들 간의 상호연관을 나타내는 공정표와 layout은 system의 state transition function(δ)이 되고, ② 일정 계획은 external event인 작업지시(X)를 발생시키는 input segment set(Ω)이 된다. Zeigler는 “experimental frame”의 개념으로 simulation experiment의 대상이 되는 부분은 data file로 분리시킬 것을 주장하고 있다 [23]. 앞의 model 구성요소들($S, X, \delta, \Omega, \tau$)중 어느 부분을 experimental frame으로 분리할 것인가 하는 것은 그 simulation study의 목적에 달려 있다고 볼 수 있다. 본 연구의 대상인 대량 생산체제의 simulation의 경우 다음과 같은 experiment를 생각할 수 있다.

- ① 설비 용량의 결정
- ② 공장 layout 결정
- ③ 제품 공정의 선택
- ④ 생산 일정계획의 평가

만일 ④번의 목적만을 위해 simulation experiment를 하는 경우라면 생산 일정 계획만을 data file로 분리시켜도 충분할 것이다. 본 연구에서는 앞의 4가지 목적 모두를 실험할 수 있도록 file structuring 하는 경우를 가정하여 그 연관을 <그림 11>에 나타내었다.

7. 결론 및 고찰

다품종 대량생산체제는 line 화에 의한 대량 생산방식의 잇점을 살리면서 line-change-over에 따른 재공재고를 줄일 수 있도록 설계되고 운용되어야 한다. 이를 위하여는 공장의 초기 설계단계에서는 물론 공장운용단계에서 체계적인 시뮬레이션 분석이 필수적이다. 오늘날 대부분의 대량생산체제가 다품종 생산을 겨냥하고 있으므로 이러한 시스템에 대한 체계적인 모델링 방법의 제시는 매우 중요한 의미를 갖는다. 또한 본 논문에서 제시된 모델링 방법론의 많은 부분은 대량 생산체제 이외의 다른 형태의 제조 system에도 적용될 수 있을 것으로 기대된다.

본 논문에서는 다품종 대량생산 체제의 모델링에 있어서

- ① discrete-event formalism, activity 중심 world-view의 선정,
- ② ACD에 의한 component modeling-system modeling의 단계적 과정
- ③ program 구성을 위한 file structuring의 방향 제시

등을 밝히고 있다. 본 논문에서 제시한 절차에 맞추어 앞서 예로 들은 D 회사 가전제품 공장을 모델링 하여, Pascal language로 program을 작성하여 simulation 해 보아 성공적인 결과를 얻을 수 있었다.

궁극적으로는 시스템에 대한 모델이 얻어진다면 이 결과가 바로 simulation program으로 변환될 수 있어야 한다. ACD를 사용하여 system을 기술하는 경우 ACD가 bipartite graph인 경우에는 자동적으로 simulation program으로 변환시킬 수 있다. 그러나 본 논문에서 다룬 대량생산체제 또는 기타의 제조 system을 보다 상세하게 기술하기 위해서는 bipartite graph가 아닌 일반적인 형태의 ACD가 필요하다. 이를 위한 ACD syntax의 정립과 그에 대한 automatic program generation

에 관한 연구가 요구된다. 또한, 여러 형태의 system에 대해 ACD를 적용해서 구성요소 model library를 구축하면 system 분석가들에게 많은 도움이 될 수 있으리라 생각된다.

본 연구에서는 modeling 문제를 중심으로

structuring까지만 다루었으나 programming 문제도 많은 관심을 갖는 연구 분야가 되리라 생각되어 후일 이 문제에 관한 연구가 본 논문과 연관되어 진행될 것을 기대한다.

References

1. Bevans, J. P., "First, Choose an FMS Simulator," *American Machinist*, May 1982, pp. 143-145.
2. Bollinger, J. G., and J. R. Crookall, "The Role of Simulation in Design/Teaching of Manufacturing Systems," *Annals of CIRP*, Vol. 30, No. 2 (1981), pp. 525-532.
3. Boothroid, G., C. Poli and L. E. Murch, *Automatic Assembly*, Marcel Decker, Inc., 1983.
4. Browne, J., and K. Rathmill, "The Use of Simulator as a Design Tool for FMS," *Proceedings of 2nd Conference on FMS*, 1983, pp. 185-196.
5. Dornhoff, L. L., and F. E. Hohn, *Applied Modern Algebra*, McMillan, 1978.
6. Hutchinson, G. K., "The Automation of Simulation," *1981 Winter Simulation Conference Proceedings*, edited by T. I. Oren, C. M. Delfosse, and C. M. Shub, pp. 489-495.
7. Law, A. M., and W. D. Kelton, *Simulation Modeling and Analysis*, McGraw-Hill, 1982.
8. Mathewson, S. C., "The Application of Program Generator Software and Its Extensions to Discrete Event Simulation Modeling," *IIE Transactions*, Vol. 16, No. 1 (Mar. 1984), pp. 3-18.
9. Nance, R. E., "The Time and State Relationship in Simulation Modeling," *Communications of the ACM*, Vol. 24, No. 4 (Apr. 1981), pp. 173-179.
10. Oren, T. I., "Concepts for Advanced Computer Assited Modeling," in *Methodology in Systems Modeling and Simulation*, edited by B. P. Zeigler, North Holland, 1978, pp. 29-55.
11. Oren, T. I. and B. P. Zeigler, "Concepts for Advanced Simulation Methodologies," *Simulation*, Mar. 1979, pp. 69-82.
12. Pegden, D., *Introduction to SIMAN*, Systems Modeling Corp., 1982.
13. Pegden, D., and Inyong Ham, "Simulation of Manufacturing System Using SIMAN," *Annals of CIRP*, Vol. 32, No. 1 (1982).
14. Peterson, J. L., *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, 1981.
15. Phillips, D. T., and A. Alan B. Pritsker, "GERT Network Analysis of Complex Production Systems," *International Journal of Production Research*, Vol. 13, No. 3 (1975), pp. 223-237.
16. Pritsker, A. Alan B., *Introduction to Simulation and SLAM*, John Wiley & Sons, 1984.
17. Schriber, T. J., *Simulation Using GPSS*, John Wiley & Sons 1974.
18. Schroer, B. J., J. T. Black, and Shou Xiang Zhang, "Just-in-Time, with Kanban, Manufacturing System Simulation on a Microcomputer," *Simulation*, Aug. 1985, pp. 62-70.
19. Schruben, L., "Simulation Modeling with Event Graph," *Communication of the ACM*, Vol. 26, No. 11 (Nov.1983), pp. 957-963.
20. Spriet, J. A., and G. C. Vansteenkiste, *Computer Aided Modeling and Simulation*, Academic Press, 1982.

21. Torn, A. A., "Simulation Graphs: A general Tool for Modeling Simulation Designs," *Simulation*, Dec. 1981, pp. 187-194.
22. Torn, A. A., "Simulation nets, a Simulation Modeling and Validation Tool," *Simulation*. Aug. 1985, pp. 71-75.
23. Zeigler, B. P., "System Theoretic Representation of Simulation Models," *IIE Transactions*, Vol. 16, No. 1, pp. 19-34.
24. Zeigler, B. P., *Theory of Modeling and Simulation*, John Wiley & Sons, 1976.