

Petri Nets를 사용한 유연생산체제(FMS)에서의 의사결정지원체계(DSS)설계 Modelling the Decision Support System in Flexible Manufacturing Systems Using Petri Nets

김진규*

ABSTRACT

Contingency response in a FMS system requires that the system be able to identify and evaluate a number of alternatives. This paper is outlined that DSS completes with a problem processor in incorporating Timed Petri Nets model, and a data base system. DSS enables the FMS user to get the maximum benefit from a FMS. The structure of this DSS parallels the organizational activities involved in running the FMS.

In designing a shop floor controller for FMS, interactions between the different real time, discrete event functions must be established. The specification of the machine tool and material handling system functions working as a concurrent and cooperative system is given in terms of Timed Petri Nets. In addition, the basic concepts and uses of Petri Nets are surveyed.

The system to be studied is first modelled by a Timed Petri Nets and then procedure for evaluating the FMS system performance are presented. Numerical examples are studied to illustrate a session of the FMS used-DSS interaction.

*한양대 산업공학과 박사과정

1. 서 론

오늘날 제품의 수요는 점차 다양화되고 그 제품의 생산수명(life cycle)이 점점 짧아진 추세에서 생산체계의 유연성과 생산성의 문제를 해결하기 위해서 생산제조공장의 자동화 일환으로 유연(자동) 생산체계(Flexible Manufacturing Systems ; FMS)가 등장하였다[2, 7, 15]. 이 FMS는 생산제품에 따라 유연성 있게 제조라인을 운영하여 생산성을 높일 수 있는 자동생산시스템으로서, 자동화된 자재운송시스템(Material Handling System ; MHS)에 의해 연결된 자동화된 NC기계들의 생산제조시스템이다.

FMS의 목적은 다품종소량 생산시스템의 유연성과 자동화된 대량시스템의 생산성을 동시에 추구하는 것이다. 이 목적을 실현하기 위해서 FMS는 실제적인 절삭가공작업, 부품의 이동, 공구의 교환등과 같은 실시간(real time)활동들을 감시하고 제어하는데 아주 많은 컴퓨터 하드웨어와 소프트웨어가 요구 되어진다. 이와 같은 FMS의 복잡성 때문에 이 시스템을 구현하기 위해서는 시스템이 제대로 동작하도록 조정하는 제어체계가 구성되어야 한다. 제어체계는 대개 몇 단계로 구성되는데, 이 단계들 사이의 정보와 제어의 흐름이 FMS구축에 매우 중요한 요소가 된다[6].

진보된 FMS가 세계도처의 생산공장에 설치되는 동안에도 이 시스템들의 능력은 대체적으로 완전히 발휘되지 못해왔다. 그 이유는 FMS의 효율적인 운영은 매우 유능한 생산공정 관리자에게도 매우 어렵기 때문이다. 어떤 특정한 자원을 한 생산제품에 할당하는 의사결정은 모든 다른 제품을 생산하는데 이용할 수 있는 다른 자원에 영향을 미칠 뿐만 아니라, 이들의 상호작용(interaction)은 매우 복잡하여 예측하기

도 쉽지 않다. FMS는 매우 값비싸기 때문에 가능한한 효율적으로 설계하고 운영하는 것이 매우 중요하다. 이와 같은 목적을 실현하기 위해서 컴퓨터에 기초한 의사결정지원체계(Decision Support System ; DSS)을 구축하여야 한다[18].

Ravichandran과 Chakravarty(1980) 그리고 Suri와 Whitney(1984)는 FMS에서의 DSS는 매일 매일의 운영단계(operational phase)에서 다양한 대체적인 활동의 결과를 평가하기 위한 FMS사용자를 돕기 위한 도구라고 정의하였다. 또한 Nof들(1980)은 제조운영체계(manufacturing operation system)를 자동화된 제조의 통제와 의사결정지원체계로서 표현했는데, 이때에 세가지의 기본모델은 참고자료(reference data), 운영자료와 의사결정논리(decision logic)인데 이들을 Petri Nets(PN)의 아류인 E-Net(Evaluation Net)으로 표현되었다.

DSS는 근본적으로 통제시스템을 수행하기 위한 수단인데, 의사결정 모듈 선택이 자동화되어 거의 인공지능(artificial intelligence)에 가까운 능력을 보유하게 되거나, 아니면 DSS는 단지 대체적인 조치의 대안만을 제시해주며 최종선택은 인간 사용자 즉 감독운영자에게 위임하여 컴퓨터와 인간 의사결정자와 대화형(interactive)으로 처리하는 시스템을 제공하는 것이다.

따라서 본 연구에서는 DSS의 서브시스템인 문제처리시스템(Problem Processing System ; PPS)을 Timed Petri Nets(TPN)의 개념을 사용하여 구축하고자 한다. 그리고 FMS에서의 다양한 의사결정 대체안을 TPN을 사용하여 모형화한 다음, 그 제조 수행도를 평가한다.

본 연구의 내용 구성은 제2장에서는 FMS의 DSS를 설명하여, 제3장에서는 TPN을 사용한 FMS를 모형화한다. 제4장에서는 제3장에서 모

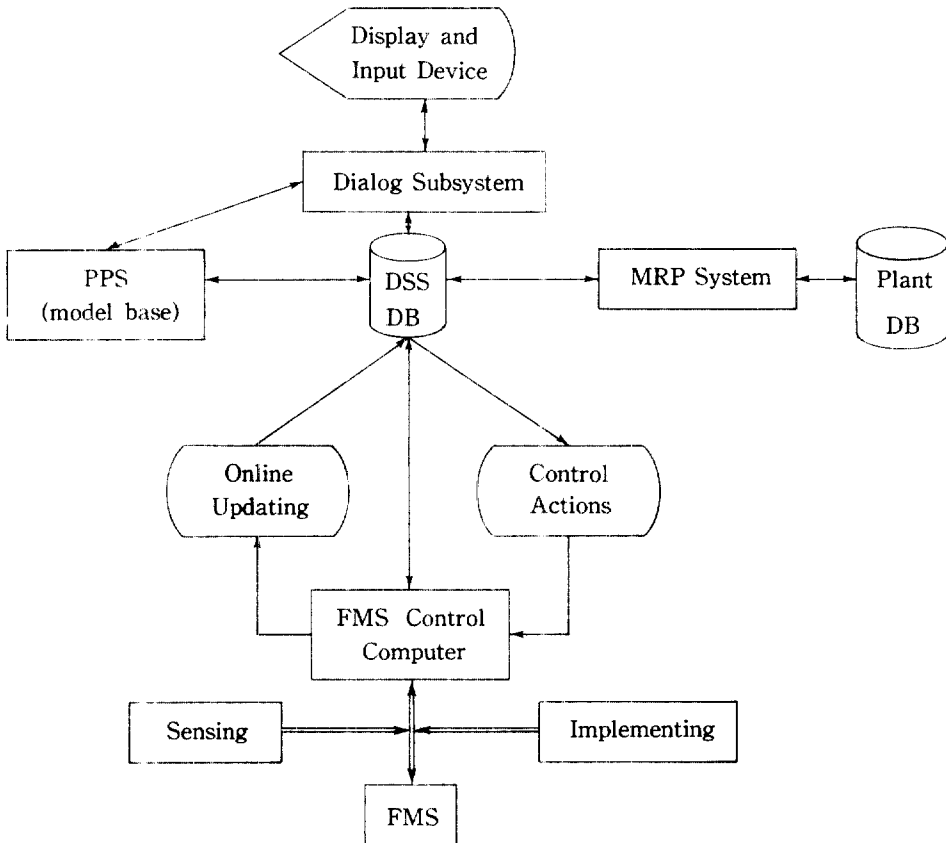
형화한 FMS의 수행도를 평가하며, 마지막으로 결론을 유도한다.

2. FMS에서의 의사결정지원체계

FMS에서의 DSS구조는 다음 <그림1>과 같이 나타낼 수 있다[15].

FMS는 FMS통제컴퓨터에 의하여 감시되며, 이는 실시간 감지의 도움을 통하여 FMS에서의 다양한 자원(resources)들의 위치를 FMS DB

에 갱신 유지시킨다. DSS는 제조요구계획시스템(Manufacturing Requirements Planning System)을 통하여 설비 DB와 연결되며, FMS 컴퓨터는 FMS에서의 통제활동 즉, part프로그램의 선적(loading)을 개시할 수 있다. DSS DB는 모델베이스를 통한 대화형 서브시스템(dialog subsystem)과 서로 계면(interface)된다. 모델베이스는 PPS로 활동하며, 대화형 서브시스템을 FMS사용자의 질의와 명령을 PPS로 번역한다. PPS는 FMS통제컴퓨터와 FMS DB에 작동한다.



<그림 1> DSS structure of FMS system

FMS운영의 의사결정구조는 적절한 하드웨어와 소프트웨어를 사용하여 어떻게 FMS운영의 사결정을 수행하느냐를 보여준다. FMS의 성공적인 기능 수행은 조직체의 모든 수준에서의 활동을 계속 요구할 것이기 때문에, 그 전체의 맥락에서 FMS를 고찰하는 것이 매우 효율적이다. 요구되어지는 다양한 활동은 조직체 운영의 전통적인 3단계 수준의 견지에서 이해되어야 한다. 각 수준에서의 세부적인 내용을 표로 나타내면 <표 1>과 같다.

3. TPN을 사용한 FMS 모형화

FMS모형화(modeling)방법은 크게 시뮬레이

션 방법과 해석적인 방법으로 나뉘어진다[19]. 이중 시뮬레이션은 민감도분석(sensitivity analysis)을 통한 각 대개변수의 분석에 관련되어 시스템 수행도 예측에는 효율적이거나 아주 많은 입력자료를 요하므로 대화형(interactive)인 DSS의 경우에는 비효율적이다. 또한 대기행렬 네트워크(queueing network)와 같은 해석적 방법은 FMS 수행도 연구에 다른 체제를 제공해주는 것은 하나, 실제로 충족될 수 없는 몇가지 가정사항에 기초하므로 현실성이 결여된다.

따라서 본 연구에서는 최근에 여러분야 즉 다중처리, 분산처리, 하드웨어와 소프트웨어 시스템, 제조시스템등의 모형화에 다양하게 사용되고 있는 PN을 확장한 TPN을 사용하여서 효율적인 FMS의 DSS를 모형화 하고자 한다.

<표 1> Levels of FMS operational decision making.

Time Horizon	Management Level	Typical Tasks	Typical DSS Software	Hardware
Long Term (Months/ Years)	Upper Management	<ul style="list-style-type: none"> Part mix changes System modification/ expansion 	<ul style="list-style-type: none"> Part selection program Queuing model Simulation 	<ul style="list-style-type: none"> Mainframe computer or DSS computer
Medium Term (Days/Weeks)	FMS Line Supervisor	<ul style="list-style-type: none"> Divide production into batches Maximize machine utilization Respond to disturbances in production plan/ material availability 	<ul style="list-style-type: none"> Batching and balancing programs Simulation 	<ul style="list-style-type: none"> DSS computer or FMS computer
Short Term (Minutes/ Hours)	FMS Line Supervisor	<ul style="list-style-type: none"> Work order scheduling and dispatching React to system failure Tool management 	<ul style="list-style-type: none"> Work order dispatching, program Operation and tool reallocation program Simulation 	<ul style="list-style-type: none"> FMS computer

3.1 Timed Petri Nets

PN은 1965년 Petri, C.A.에 의하여 고안된 비동기적(asynchronous)이고 동시(concurrent)에 일어나는 정보의 흐름 및 제어를 방향(directed)의 그래프 형식으로 나타낸 것이다 [1, 12, 13, 17].

1) PN의 정의

PN은 방향성의 그래프로 다음과 같이 정의된다.

$$PN : N = (T, P, A) \dots\dots\dots (1)$$

$T = \{t_1, t_2, \dots\dots\dots t_n\}$; 변이(transition)의 집합

$P = \{p_1, p_2, \dots\dots\dots p_n\}$; 위치(place)의 집합

$N = \{TUP\}$

$A \subseteq \{T * P\} \cup \{P * T\}$; 방향성 arc의 집합

PN의 표시(mark) M 은 다음과 같이 나타낸다.

$$M : P \rightarrow I \text{ 단, } I = \{0, 1, 2, \dots\dots\} \dots\dots\dots (2)$$

M 은 Net에서 각 위치에 토큰(token; 정수값)을 할당하며, 시스템에서의 전개(evolution)을 나타낸다. M_i 는 p_i 는 할당된 토큰의 수를 나타내며 이러한 PN을 Marked PN이라고 한다.

$$\text{Marked PN} : C = (T, P, A, M) \dots\dots\dots (3)$$

그래프에서 동심원(circle)은 위치를 나타내며, 선(bar)은 변이를 나타내며, 점(dot)은 토큰을 나타낸다. PN에서 위치는 프로그램 수행

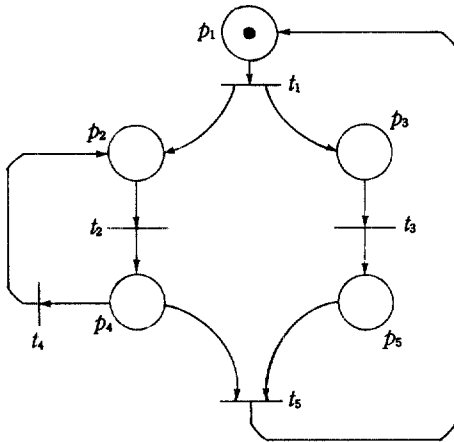
을 위한 대기조건을 나타내며, 변이는 실시스템에서 사건의 출현(발생)을 나타내며, 토큰은 이 사건들의 출현횟수에 대응된다. 표시 M 을 가진 PN은 사건들의 조절이나 동기화(synchronous)를 시뮬레이션하며, 토큰의 전개는 fire하기 위한 변이의 조건을 만들며 또한 fire에 의한 조절된 조건을 나타낸다.

Marked PN $C = (T, P, A, M)$ 에서 모든 투입위치 $I(t) = \{p | (p, t) \in A\}$ 에 대하여 $M(p) > 0$ 이면 변이 t 는 fire가 가능하다(enabled)고 한다. 각 투입위치로부터 토큰을 제거하여 각 산출위치 $O(t) = \{p | (t, p) \in A\}$ 로 옮기는 것을 firing이 가능한 변이 t 가 fire되었다고 한다. 이 결과는 M' 와 같다.

$$M'(p) = \begin{cases} M(p) + 1 & \text{if } p \in O(t) \\ M(p) - 1 & \text{if } p \in I(t) \\ M(p) & \text{otherwise} \end{cases} \dots\dots\dots (4)$$

PN의 가장 중요한 모형화 성질은 동시화(concurrency)와 충돌(conflict)를 나타낼 수 있는 능력이다. <그림 2>에서 t_1 은 fire할 수 있는 유일한 변이이며 이것이 완료되면 p_1 은 비게 되고 p_2 와 p_3 는 토큰으로 차게된다. 이 단계에서 t_2 와 t_3 는 모두 enable되며 어떤 투입위치도 공유하지 않기 때문에 동시(concurrently)에 fire할 수 있다.

PN을 표현하는데 이와같은 그래픽 방법 이외에도 최근에는 논리적인 표현방식(rule-based interpretation)을 사용하기도 한다. Daliang (1989)은 전문가 시스템(expert system)의 언어인 PROLOG를 이용한 PNP(Petri Nets in PROLOG)를 개발했다. 이 PNP로서 <그림 2>의 PN를 나타내보면 아래와 같다.



where ;
 ○ = place (condition)
 - = transition (event)
 • = token (enabled condition)
 ↓ = arc (control flow)

<그림 2> PN example of concurrency

Place(p_1) ; Place(p_2) ; Place(p_3) ;
 Place(p_4) ; Place(p_5) ;
 Token(p_1) ;
 Transition(t_1 , in $\cdot p_1$, out $\cdot p_2$, out $\cdot p_3$) ;
 Transition(t_2 , in $\cdot p_2$, out $\cdot p_4$) ;
 Transition(t_3 , in $\cdot p_3$, out $\cdot p_5$) ;
 Transition(t_4 , in $\cdot p_4$, out $\cdot p_2$) ;
 Transition(t_5 , in $\cdot p_4$, in $\cdot p_5$, out $\cdot p_1$) ;

<그림 3> Rule-based interpretation
 by PNP of <그림 2>.

<그림 2>에서 만약 토큰이 p_4 와 p_5 에 차서 t_4 와 t_5 가 둘다 enable 되었으나 하나의 fire에 다른것의 fire를 불가능하게 한다. 즉, t_4 가 fire하면 t_5 는 disable되며, t_5 가 fire하면 t_4 는 disable된다. 이런 상황을 충돌(conflict)이라고 한다.

Marked PN $C = (T, P, A, M)$ 에서 모든 $M' \in R(M)$ 에 대하여 $M'(p) \leq k$ 이면 이 Net는 k -한계(bounded)라고 한다. 여기서 $R(M)$ 은 M 으로부터 접근(reachable)할 수 있는 모든 표시

의 집합이다. 만약 1-한계이면 이 Marked PN은 안전(safe)하다고 하며, 변이 t 가 enable된 곳에 M' 로부터 도달할 수 있는 표시가 존재할 때에 이 Net는 생명력(liveness)이 있다고 한다.

PN을 분석하는 기본 수단으로는 접근가능성 나무를 사용하는 것이다. Marked PN C 의 접근가능성 나무의 노드(node)는 Net에서 접근가능성이 있는 표시를 나타낸다. C 에 대한 접근가능성나무 형성의 체계적이고 효율적인 알고리즘은 IBM의 Agerwala(1979)에 의하여 제시되었다.

2) TPN

본래의 PN은 시간의 척도가 없는 Untimed PN으로서 변이는 순간적으로 일어났다. 그러나 시간의 중요한 성질은 사건출현의 순서를 규정하는데 있기 때문에 PN실행은 이산적인 사건을 표현하는 변이의 시간적 순서를 통하여 시스템 행위를 모형화 한다. 변이의 firing은 '0'의 시간을 갖는 순간적인 사건으로 간주되어 원시

적인 사건(primitive event)이라고 한다.

최근에는 시간을 고려한 Timed PN을 이용하여 많은 연구를 하고 있으며 특히 시간개념도 결정적(deterministic)에서 확률적(stochastic) 개념으로 바뀌어 가고 있다. Ramamoorthy와 Ho(1980)는 fire하는 시간을 가진 각 변이를 고려한 TPN의 수행도 평가를 하였다. Kamath와 Viswanadham(1986)은 FMS에 있어서 시뮬레이터 설계와 하드웨어와 소프트웨어 설계에 사용된 논리증명에 있어서 TPN의 적용방법을 개론형식으로 제시하였다. 그리고 Barad와 Sipper(1988)도 TPN을 사용하여 생산자동화에서의 유연성(flexibility)을 묘사하였으며, 이밖에도 이에 관련된 많은 연구가 있다[4, 8, 9, 20].

따라서 본 연구에서도 원래의 PN의 개념에 생산시간의 개념(즉, 가공시간, 준비시간, 이동시간등)을 고려한 TPN을 이용하여 FMS를 모형화한 다음, 이의 효율적인 DSS구축을 하도록 하겠다.

FMS와 같은 분산처리나 온라인의 실시간 시스템의 설계에 이 TPN을 사용하는 데는 다음과 같은 장점을 가지고 있다.

첫째, 시스템 전체가 그래프의 특징과 표현체계의 정확한 성질때문에 매우 복잡한 시스템도 이해하기가 용이해지며, 둘째, 병렬(parallel)로 수행되는 프로그램의 행위가 접근가능성 분석도구(즉 접근가능성 나무)를 사용하여 쉽게 입증된다. 셋째, 시스템이 상하양식(top-down)으로 자세하게 모형화 될 수 있고 TPN의 동시화 성질때문에 통제가 용이해진다. 마지막으로 잘 개발된 PN분석기법에 의하여 시스템이 체계적이며 완벽한 정성적(qualitative)분석과 TPN를 사용한 시스템 수행도를 평가할 수 있다.

3.2 FMS에서의 TPN모듈

FMS의 세가지 중요한 구성요소는 1) 여러가지 기능을 수행할 수 있는 수치제어 공작기계 2) 자동화된 MHS시스템 3) 컴퓨터 하드웨어 및 소프트웨어이다. 최근에 설치 운용되고 있는 FMS들은 다음 <표 2>에 나타나 있는 작업 내용과 흐름의 실시간 활동들을 다양한 의사결정규칙에 따라서 컴퓨터로 제어한다. 그리하여 FMS관리감독자로 하여금, 효율적인 FMS의 DSS를 구축하게 하여 FMS효율을 향상시킬 수 있게 한다.

FMS에서의 생산활동은 자동화되어 있으나, 시스템 운용 방해(distruption)가 발생하여 FMS관리감독자가 시정조치를 취해야 할 사항이 발생하게 된다. 인간의 관여와 의사결정을 요하는 방해에는 기계고장, 공구고장, 공구교체 경고, AGV고장등이다. 이런 환경에서의 의사결정은 대개 비구조적(unstructure)이므로 FMS관리감독자는 효과적인 의사결정을 하기위해서 <표2>과 내용을 대화형으로 지원을 받아야 될 필요가 있다.

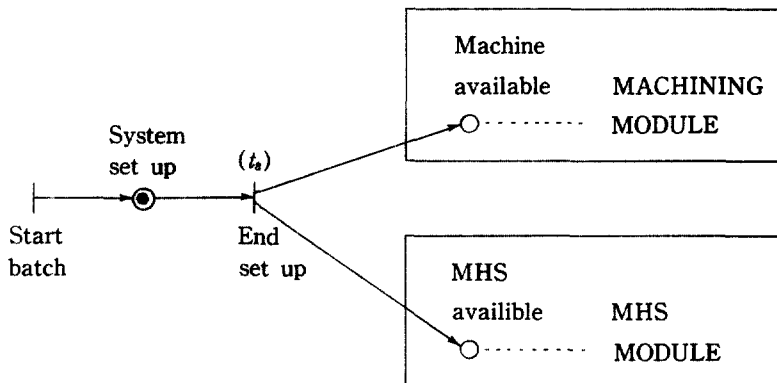
TPN의 취약점은 규모가 큰 시스템을 표현하는데 실제적인 어려움이 있다. 개념적으로는 어려움이 증가하지 않지만 Net자체가 더욱 더 복잡해 진다. 따라서 TPN의 모듈화가 절실히 요구된다. FMS의 구조를 블록화하여 묘사하는데도 가공모듈(machining module)과 자재처리모듈(material handling module)로 나누어서 그 복잡성을 줄이도록 한다. 이 두 모듈이 삽입된 FMS의 TPN모형화는 <그림 4>와 같이 나타낼 수 있다.

1) 가공 모듈

TPN을 사용한 FMS 가공모듈은 <그림 5>와

<표 2> Summary of operations and decision rules in the FMS

No	operation items selected	available rules
1	NEW PART to be released into the system	FIFO, EDD Highest Priority(HP).
2	PALLET to mount a part	Idle Pallet, Lowest Utilization, HP.
3	AGV to transport a part	Idle AGV, Nearest AGV, Lowest Utiliza- tion, Shortest Approach Time, HP.
4	MACHINE to perform a requested operation	SPTA, ESTA, EFTA, Shortest Queuing Length, HP.
5	PART from the queue of a station for processing	FIFO, LIFO, SPT, LPT, EDD, Smallest (or Largest)Slack Time, Least (or Most) Number of operations, HP.
6	TRANSPORT PATH to a next station	Predetermined Path, Shortest Traveling Path,
7	OPERATOR if necessary	Idle Operator, Nearest Operator Highest Skilled Operator.
8	PART'S NEXT OPERATION	Predetermined Operation, EST, HP.



t_s : system set up time

<그림 4> An overall view of TPN model of the FMS

같이 나타낼 수 있다. 이 FMS에서는 기계고장으로 인한 간섭을 나타내는 TPN모형화 구조를 제시했다. 기계고장이 없는 정상작업경로는 $T_1-T_2-T_3$ 이다. 그러나 간섭이 발생할 때의 두 부분경로는 다음과 같다.

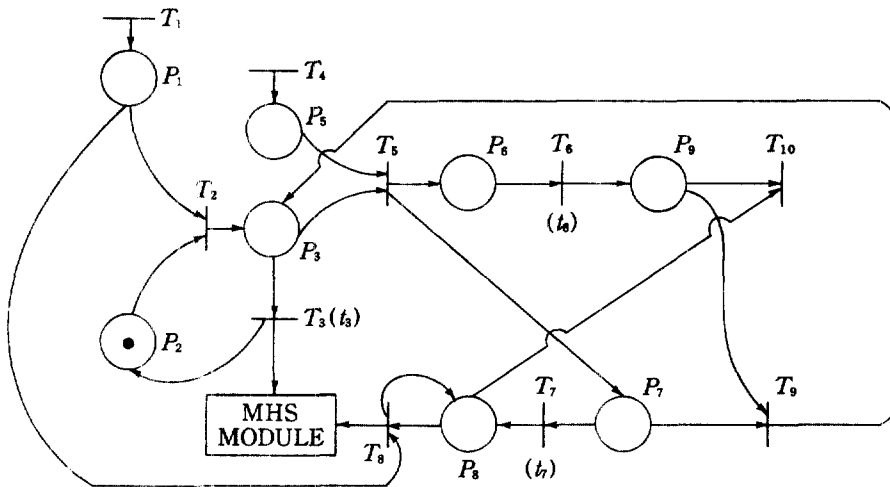
i) 고장이 발생했을 때(T_4), 경로의 유통성이 없는 시스템의 작업경로는 $T_1-T_2-T_5-T_6-T_9-T_3$ 이다. 단, 이 경우에 재 경로를 위한 조정시간 t_7 이 수리시간 t_6 보다 길다.

ii) 고장이 발생했을 때(T_4), 경로의 유통성

이 있을때 즉 대체적인 작업공정이 존재하여 비교적 재경로를 위한 조정시간 t_7 이 짧을 때에 시스템의 작업경로는 $T_1-T_2-T_5-T_7-T_8$ 이다.

2) 자재 처리 모듈

TPN을 사용한 FMS의 자재처리모듈도 역시 <그림 5>와 같이 나타낼 수 있으며, 여기에서도 정상적인 운반경로와 AGV가 고장이 났을때 유통성이 없을때와 있을 때의 부분적인 경로도 1)의, i), ii)에서와 같이 규정할 수 있다.



- | | |
|-----------------------------------------|----------------------------------------|
| P_1 : Job is waiting | T_1 : Job arrives |
| P_2 : Machine available | T_2 : Start machining |
| P_3 : Job is being proceed | T_3 : End machining |
| P_5 : Machine breakdown | T_4 : Machine failure arrives |
| P_6 : Machine repair | T_5 : Stop machining |
| P_7 : Adjusting for rerouting | T_6 : End repair |
| P_8 : Control rerouting | T_7 : Start control rerouting |
| P_9 : Machine in working condition | T_8 : Reroute |
| t_3 : Processing time | T_9 : Cancel adjusting for rerouting |
| t_6 : Repair time | T_{10} : End control rerouting |
| t_7 : Time of adjusting for rerouting | |

<그림 5> MACHINING MODULE of a FMS

4. FMS에서의 수행도 평가

TPN을 사용한 FMS의 수행도 평가는 이 수행도평가 방법이 아주 제한된 부류의 PN에 국한되어 이용가능하기 때문에 최근까지 많은 관심을 끌지 못하였다[5]. 그리하여 단지 시스템에 대한 통제부분만이 시스템 수행도 평가에 이용될 수 있었다[14].

본 연구에서의 FMS 수행도 평가 문제는 FMS의 최대의 수행도, 즉 FMS의 작업처리에 있어서 최소의 생산가공 주기(production cycle)를 구하는 것이다.

TPN에 의하여 모형화된 FMS는 일치시스템(consistent system)과 불일치 시스템(inconsistent system)으로 크게 나누어진다. 대부분의 실제 생산 환경에서의 시스템은 전자의 경우에 속하는데, 이는 또다시 의사결정용이시스템(decision free system), 안전시속시스템(safe persistent system), 그리고 일반시스템(general system)으로 나누어진다. 이 각자의 종류에 따라서 분석방법과 복잡성(complexity)이 다르다.

만약에 모든 위치에서 투입되는 변이에 할당된 정수의 합과 산출되는 변이에 할당된 정수의 합이 같은 '0'이 아닌 정수가 존재한다면 이 TPN은 일치(consistency)라고 하며, 이 Net로 모형화된 FMS도 일치시스템이라고 한다. 만약 그렇지 않으면 이 TPN을 불일치(incosistency)라고 하며, 그때의 FMS를 불일치시스템이라고 한다. 시스템이 생명력(liveness)과 일치하면 이 시스템은 각 주기 후에 초기의 상태로 되돌아 오며 그 후에도 계속 이 주기가 반복된다. 불일치하게 되면 토큰의 무한한 갯수(즉 무한한 자원)가 산출되거나 소비되어 결국은 시스템이 정지하게 된다. 대부분 실제 생산 환경에서는

유한한 자원을 이용하고 있기 때문에 FMS도 일치시스템에 속한다. 따라서 본 연구에서도 일치시스템의 세종류에 대한 수행도 평가 방법을 다룬다.

4.1 의사결정용이시스템

각 위치에 있어서 하나의 입력 arc와 하나의 출력 arc만이 존재할 때 이 TPN을 의사결정용이(decision free)하다고 하며, 이 때의 시스템을 의사결정용이시스템이라고 한다. 이 의미는 주어진 위치에서 토큰이 미리 정해진 변이에 의해서 발생되고 소비된다는 것이다. 그러므로 이 의사결정용이의 TPN에서 토큰의 수는 어떤 firing의 순서 후에도 똑같은 갯수를 갖는다.

<정의>

변이의 T_i 가 n_i 번째 실행을 개시했을 때의 시간을 $S_i(n_i)$ 라고 두면 변이 T_i 에서의 주기 C_i 는 다음과 같다.

$$C_i = \lim_{n_i \rightarrow \infty} \frac{S_i(n_i)}{n_i} \dots\dots\dots (5)$$

이 정의에 대한 의사결정용이시스템의 이론은 다음과 같다.

<이론>

의사결정용이의 TPN에서 모든 변이는 똑같은 주기를 가지며, 이 주기의 최소값(즉 최대의 수행도) C 와 $S_i(n_i)$ 는 다음과 같다.

$$C = \max\left\{\frac{ET_k}{N_k}; k=1, 3, \dots, q\right\} \quad (6)$$

$$S_i(n_i) = a_i + C \cdot n_i \quad \dots\dots\dots (7)$$

단, $ET_k = \sum_{T_i \in L_k} t_i$: 공정 k 에서 변이의 실행시간의 합

$N_k = \sum_{P_i \in L_k} M_i$: 공정 k 에서 위치에서의 토큰의 총수

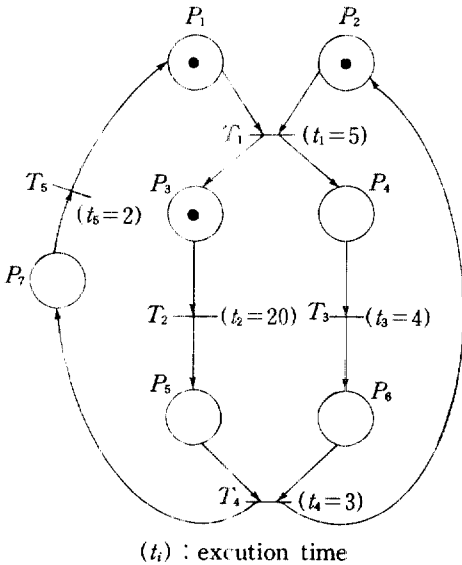
q : Net에서 공정의 수

a_i : 변이 T_i 에 관계된 상수

L_k : 공정 k

M_i : 위치 P_i 에서의 토큰의 수

이 이론에 대한 증명은 Ramamoorthy와 Ho (1980)에 의하여 이루어졌다. 이 이론에 대한 수치 예를 <그림 6>과 같은 TPN를 통하여 최소 주기를 구하면 다음과 같다.



<그림 6> Example of decision free TPN

공정 4(=k)개에 대한 주기는 다음과 같다.

i) 공정 1 : $P_1 - T_1 - P_3 - T_2 - P_5 - T_4 - P_7 - T_5$

$$C_1 = \frac{ET_1}{N_1} = \frac{5+20+3+2}{2} = 15$$

ii) 공정 2 : $P_1 - T_1 - P_4 - T_3 - P_6 - T_4 - P_7 - T_5$

$$C_2 = \frac{ET_2}{N_2} = \frac{5+4+3+2}{1} = 14$$

iii) 공정 3 : $P_2 - T_1 - P_3 - T_2 - P_5 - T_4$

$$C_3 = \frac{ET_3}{N_3} = \frac{5+20+3}{2} = 14$$

iv) 공정 4 : $P_2 - T_1 - P_4 - T_3 - P_6 - T_4$

$$C_4 = \frac{ET_4}{N_4} = \frac{5+4+3}{1} = 12$$

<그림 6>의 TPN에서 모든 공정을 열거하여서 구한 최소 주기는 15이다.

4.2 안전지속시스템

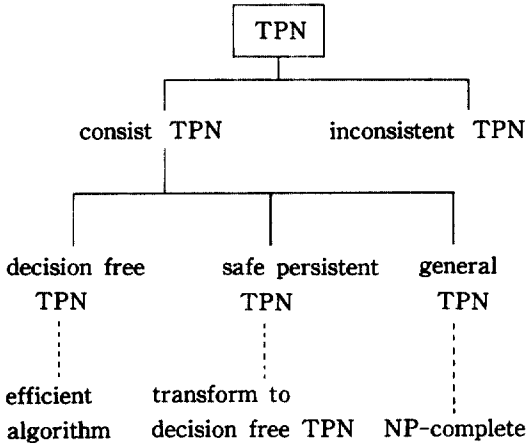
TPN이 안전(safe)하고 모든 접근 가능한 표시에 대하여 변이가 단지 변이의 firing에 의해서만 disable 될때에 이 Net을 안전지속이라고 하고 이러한 모형을 안전지속시스템이라고 한다. 이 시스템이 의사결정용이시스템과 다른점은 입력 arc와 출력 arc가 하나 이상이 될 수도 있다는 점이고 그 이외의 성질은 모두 동일하다. 이 시스템의 수행도를 계산하기 위해서는 먼저 이 시스템을 의사결정용이시스템으로 변환한 다음, 앞 절에서 논의한 방법으로 수행도를 구한다.

4.3 일반시스템

TPN이 일치하고 변이의 firing이 어떤 다른 변이에 의해서 disable되는 접근 가능한 표시가 존재할 때의 Net이 일반시스템의 PN이나, 이 Net는 의사결정용이시스템으로의 변환도 불가능하고 수행도 평가 계산량도 아주 복잡하여 이 NP-Complete문제임이 증명되었다[14].

이상의 4장에서 다룬 FMS에 있어서 TPN의

수행도 평가 방법을 각 분류별로 요약하여 나무 모양의 구조로 나타내면 다음 <그림 7>과 같다.



<그림 7> TPN classification and results of performance evaluation in the FMS

최근에는 Stochastic Petri Nets(SPN)이 컴퓨터과학 분야에 많은 관심을 받고 있다[10]. SPN는 firing시간이 지수(exponential)이거나 특정한 형태의 분포에 따르는 TPN이다. 이 중에서 특히 지수적인 분포를 하는 변이율(transition rate)를 가진 SPN모형은 Markov모형에 근거를 두고 수행도를 평가한다.

5. FMS의 DSS수행도 평가에

본 예제에서 다루는 FMS는 두 종류의 제품 형태(J_1, J_2)를 가공할 수 있는 세대의 자동제어 기계(M_1, M_2, M_3)와 선적/하역 장소(L/U), 그리고 각 제품형태에 대하여 하나의 파레트와 치구를 가진 MHS으로 구성되어 있다.

이 FMS에서 가공되는 제품조합은 균형을 이

루며, 또한 가공시간에 비해서 운반시간이 미세함으로 무시한다. 그리고 공정간 재고는 없다고 가정한다. 두 종류의 제품에 대한 공정순서는 J_1 이 $M_1 \rightarrow M_2$, J_2 가 $M_1 \rightarrow M_2 \rightarrow M_3$ 로서 각 가공시간은 다음 <표 3>과 같다. 단, 시간 단위는 분이다.

<표 3> Processing times of jobs

Job \ M	M_1	M_2	M_3
J_1	100	150	-
J_2	50	100	150

이 FMS에서 각 제품의 흐름형태를 TPN을 사용하여 모형화하면 다음 <그림 8>과 같이 나타낼 수 있다.

대체적인 의사결정은 TPN모형에 여분의 변이(dummy transition)를 도입함으로써 모형화할 수 있다. 이때의 변이시간(t_d)은 '0'이다. 본 예제에서 J_1 의 대체적인 가공순서와 시간은 다음 <표 4>와 같다.

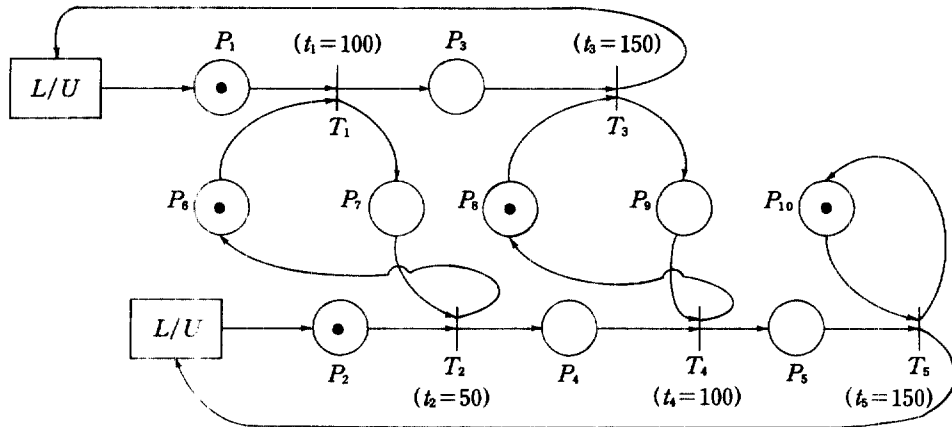
<표 4> Decision alternatives of processing J_1

DA* \ M	M_1	M_2	M_3
A	100	150	-
B	-	150	200
C	100	-	200

DA* : Decision alternative types

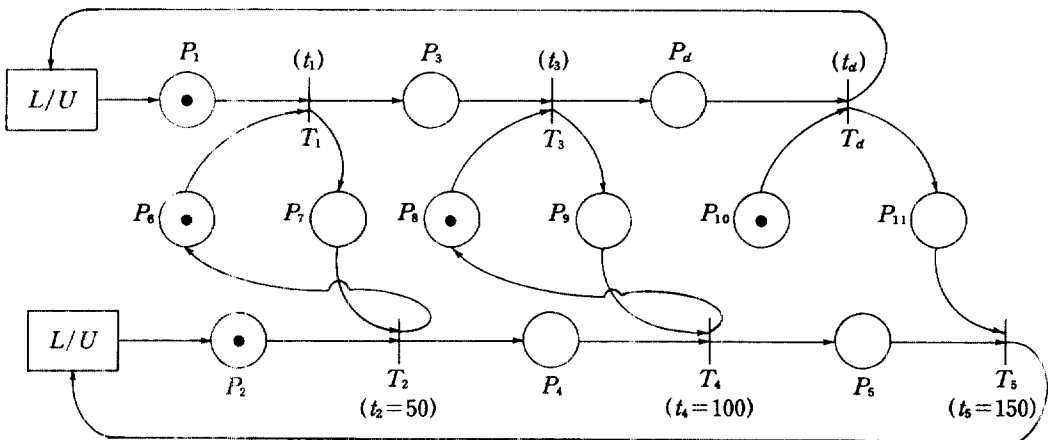
이 대체적인 의사결정을 TPN을 사용하여 모형화하면 <그림 9>와 같다.

만약 M_2 에서 J_1 을 가공하는 기계공구가 고장이 났다면, FMS 관리감독자는 DSS의 DB를



P_1 : Token for J_1 P_6 : Token for M_1
 P_2 : Token for J_2 P_8 : Token for M_2
 P_{10} : Token for M_3

<그림 8> TPN modelling of the FMS



$P_1, P_2, P_6, P_8, P_{10}$: Same mean of <그림 8>.
 P_d : Dummy place for decision alternatives
 T_d : Dummy transition for decision alternatives
 $t_1 = t_3 = t_d = x + y + z$

<그림 9> TPN modelling of decision alternatives

이용하여 시정조치를 취해야 한다. <그림 1>에서 FMS통제컴퓨터는 이 기계공구의 여분이 없으면 대체가 가능한 대체가공경로를 <표 4>와 같이 제시하게 된다. 그러면 FMS관리감독자는 이용 가능한 대체적인 가공경로의 재일정계획에 관한 의사결정을 하면된다. <그림 9>에서 가공시간 t_1, t_2, t_3 의 표현은 $x+y+z$ 로서 융통성 있게 되어 있다. 예를 들면 대체안이 A인 경우, M_1, M_2, M_3 에 대해서 아래와 같은 가공시간을 구성한다.

$$t_1 = x + y + z = 100 + 0 + 0 = 100$$

$$t_2 = x + y + z = 0 + 150 + 0 = 150$$

$$t_3 = x + y + z = 0 + 0 + 0 = 0$$

이와 같은 방식으로 대체안 B, C인 경우에도 다양하게 표시할 수 있다.

6. 결론

FMS를 효율적으로 운영하는 것은 대단히 어렵으나, FMS가 매우 값비싼 점을 감안하면 효율적인 운영은 필수적인 요청이라 하겠다. 본 연구에서는 이 FMS의 효율을 최대화하기 위하

여 집적된 FMS운영시스템의 DSS를 모형화하였다. FMS에서의 적절한 통제기능의 상하향식 설계방법이 채택되어 기계공구와 자재처리시스템, 사용자 사이에서의 일치화와 협조등의 동기화가 3단계수준으로 TPN을 사용하여서 개발되었다.

또한 PN에 대한 기본적인 정의와 FMS를 효율적으로 모형화할 수 있는 확장된 TPN의 개념을 간단히 설명하였다. 그리고 이 TPN의 성질을 이용한 고유의 시스템 평가 방법도 개개의 Net특성에 따라서 분석하였다. 마지막으로 본 연구에서 다룬 전반적인 내용을 간단하면서도 실제적인 FMS시스템을 사용하여서 예증하였다.

앞으로의 연구 과제는 일반적인 PN와 대기행렬 네트워크를 체계적으로 통합하여 PN의 두 한계인 모형화와 의사결정능력(power)을 향상시켜야 하겠다. 그리하여 제한된 저장소(buffer), 자원의 동시 점유(즉, 부품이 동시에 파레트, 공구, 기계등을 요구하는 것)와 공유된 자원(즉, 공구와 MHS)에서 기인한 방해와 같은 FMS 모형화 문제를 다룬 보다 향상된 DSS를 구축하여야 하겠다.

參 考 文 獻

1. Agerwala, T. (1979), "Putting Petri Nets to work", IEEE Computer, December, pp. 85-94.
2. Ammons, J.C., Govindaraj, T. and Mitchell, C.M. (1989), "Decision Models for Aiding FMS Scheduling and Control", IEEE Trans. on Systems, Man, and Cybernetics, Vol. 18, No. 5, pp. 744-756.
3. Barad, M. and Sipper, D. (1988), "Flexibility in Manufacturing Systems: Definitions and Petri Net Modeling", Int. J.

- Prod. Res., Vol. 26, No. 2, pp. 237-248.
4. Daliang, D. (1989), "Rule Based Discrete System Simulation and Verification of Petri Nets", Modeling and Simulation Methodology, Elsevier, North Holland, pp. 257-265.
 5. Kamath, M. and Viswanadham, N. (1986), "Application of Petri Net based Models in the Modeling and Analysis of FMS", Proc. IEEE Int. Conf. on Robotics and Automation, pp. 312-317.
 6. Kimenia, J. and Gershwin, S. B. (1983), "An Algorithm for the Computer Control of a FMS", IEEE Trans., December pp. 353-362.
 7. Kusiak, A. (1985), "FMS: A Structural Approach", Int. J. Prod. Res., Vol. 23, No. 6, pp. 1057-1073.
 8. Martinez, J., Alla, H., and Silva, M. (1986), "Petri Nets for the Specification of FMSs", Modeling of Design of FMSs, edited by Kusiak, A., Elsevier, Amsterdam, pp. 389-406.
 9. Merabet, A. A. (1986), "Synchronization of Operations in a Flexible Manufacturing Cell: The Petri Net Approach", J. of Manuf. Systems, Vol. 5, No. 3, pp. 161-169.
 10. Molly, M. K. (1982), "Performance Analysis using Stochastic Petri Net", IEEE Trans. on Comp., Vol. C-31, No. 8, pp. 913-917.
 11. Nof, S. Y., Whinston, A. B., and Bul- lers, W. I. (1980). "Control and Decision Support in Automatic Manufacturing Sys- tems", AIIE Trans., Vol. 12, No. 2, pp. 156-168.
 12. Peteson, J. L. (1977), "Petri Nets", Computing Surveys, Vol. 9, No. 3, pp. 223-252.
 13. _____ (1982), "Petri Net Theory and the Modeling of the Systems". Prentice Hall, Englewood Cliffs, Jew Jersey.
 14. Ramamoorthy, C. V. and Ho, G. S. (1980), "Performance Evaluation of Asynchronous Concurrent Systems Using Nets", IEEE Trans. on Software Engg., Vol. SE-6, No. 5, pp. 440-449.
 15. Ranky, P. (1983), "The Design and Operation of FMS", IFS Ltd, UK, North-Holland.
 16. Ravichandran, R. and Chakravarty. A. K. (1980), "Decision Support in FMS Using Timed Petri Nets", J. of Manuf. System, Vol. 5, pp. 89-101.
 17. Reisig, W. (1985), "Petri Net Intro- duction", Springer-Verlag, New York.
 18. Suri, R. and Whitney, C. K. (1984), "Decision Support Requirements in Flex- ible Manufacturing", J. of Manuf. Sys- tem, Vol. 3, No. 1, pp. 61-68.
 19. Yao, D. D. and Buzacott, J. A. (1985), "Modeling the Performance of FMSs", Int. J. Prod. Res., Vol. 23, No. 5, pp. 945-949.
 20. Zhou, M. and DiCesare, F. (1990), "A Petri Net Design Method for Automated Manufacturing Systems with Shared Resources", Proc. IEEE Int. Conf. on Robotics and Automation, pp. 526-531.