

A Petri-Net Based Execution Model of Processing Equipment for CSCW-based Shop Floor control in Agile Manufacturing

Soondo Hong, Hyuenbo Cho, Mooyoung Jung

Deptment of Industrial Eng., Pohang University of Science and
Technology, Pohang, 790-784, Republic of Korea

ABSTRACT

A shop floor control system(SFCS), a central part of agile manufacturing, performs the production activities required to fill orders. In order to effectively control these activities, CSCW (computer supported cooperative work) is adopted where a supervisor does not exist. In this paper, we define functional perspective of CSCW-based shop floor control using planning, scheduling, and execution functions. In particular, we focus on an execution model that can coordinate the planning and scheduling functions. Execution can be defined informally as a function that downloads and performs a set of scheduled tasks. Execution is also responsible for identifying and resolving various errors whether they come from hardware or software. The purpose of this research is to identify all the execution activities and solving techniques under the assumptions of CSCW-based heterarchical control architecture. This paper also proposes a classification scheme for execution activities of CSCW-based heterarchical control architecture. Petri-nets are used as a unified framework for modeling and controlling execution activities. For solving the nonexistence of a supervisor, A negotiation-based solution technique is utilized.

Key words: CSCW-shop floor control, execution model, heterarchical control, petri-nets, negotiation-based solution technique.

1. 서론

1.1 연구 배경 및 동기

생산시스템의 주변 환경은 보다 짧은 생산 주
기에서 고품질의 제품을 생산할 것을 생산시스템에
요구하고 있다. 생산시스템은 이러한 환경하에서,
다양화되는 소비자의 구매 욕구를 만족시키며 예측

하기 어려운 시장 환경의 변화에 끊임없이 신속하
게 대처해 나가야만 생존할 수 있게 되었다. 기존의
유연생산시스템 (FMS), 컴퓨터통합생산시스템
(CIM) 등은 다양한 고객의 욕구를 만족시키고 자동
화나 생산시스템의 통합화 등에 적합한 개념은 제
시하였으나, 끊임없이 변화하는 시장 환경에서의

생존이나 세계화 추세에는 적합하지 못했다. 즉 민첩성 (agility) 으로 일축 될 수 있는 생산시스템의 조건을 만족시키는 새로운 생산시스템이 필요하게 되었다.

shop floor control system (SFCS) 은 전체 생산공정의 민첩성에 적지 않은 영향을 미친다. SFCS는 제품을 생산하기 위하여 공장 내의 장비들을 가동하며, 이에 필요한 가공자원들을 배치한다. 이러한 SFCS의 다양한 컨트롤 동작을 통합하기 위하여 세분화법 (decomposition technique) 을 사용하는데, 기존의 대부분의 시스템은 계층적 컨트롤 (hierarchical control) 을 일반적으로 사용하고 있다. 계층적 컨트롤 구조는 시스템의 이해가 용이하고, 시스템을 전체적으로 컨트롤함으로써 시스템 전체 최적화가 어느 정도 가능하다. 또한 중복 (redundancy) 을 줄일 수 있는 장점을 갖고 있다. 그러나, 컨트롤 요소 (control component) 가 많기 때문에 신속한 시스템 개발이 어려워 앞서 언급한 민첩성을 만족시키기에 부족하다. 또한 계층이 존재하기 때문에 공장의 레이아웃의 변화에 신속하게 대처하지 못하고, 한 장비의 고장이나 시스템 일부의 확장이 시스템 전체에 영향을 미치게 된다.

다른 한편으로 생산 기술은 긍정적인 방향으로 변화하였다. 컴퓨팅 코스트가 작아지고, 저가의 지능시스템이 제공되며, 네트워크 통신 기술이 끊임없이 향상되고 있다. 컴퓨팅이나 통신은 SFCS의 컨트롤 구조와 밀접한 관계를 갖고 있다. 그러나 계층적 컨트롤 구조는 컴퓨팅과 통신에 의존도가 적은 컨트롤 구조로써, 이러한 생산 환경의 긍정적인 변화를 충분히 활용하지 못했다.

1.2 연구 목적 및 내용

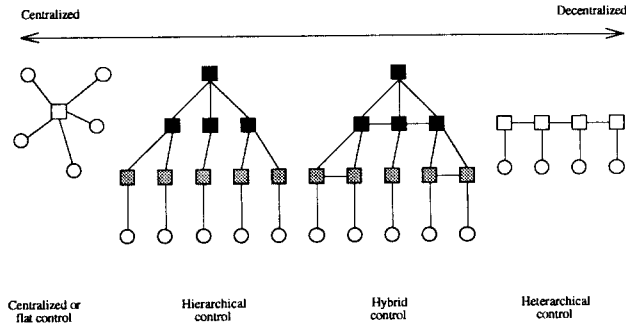
본 연구는 민첩성을 만족시키기 위하여 수평적 계층 구조 (heterarchical control architecture) 를 적용하였다. SFCS 컨트롤러를 기능상으로 의사결정 (decision making) 을 담당하는 planning과 scheduling, 그리고 메세지 모니터링과 해석, 실행

베이스 의사결정, 메세지 전송 및 장비 코드 다운로드, 협상, 고장 감지 및 진단 등을 담당하는 execution으로 나누었다. 이 중에서 planning과 scheduling에서 결정된 내용을 처리하는 execution function을 연구 대상으로 한다. 먼저 execution function을 정의하고, execution entity로 분류하고, 모든 execution activity를 명시하며, 페트리네트로 모델링하여 execution문제를 해결하고자 한다. 관리자가 존재하지 않는 수평적 컨트롤 구조 (heterarchical control architecture) 의 문제를 해결하기 위하여 컴퓨터 지원 협동작업 (computer supported cooperative work, CSCW), 협상 베이스 해법 (negotiation-based solution technique) 을 이용하였다.

2. 기존 연구 고찰

2.1 SFCS의 control architecture

SFCS는 비즈니스시스템에서 입력되어지는 제품주문서에 따라 제품을 생산한다. 제품 주문서에는 납기일자, 품질, 생산 우선도 등이 명시되어 있다. SFCS는 명시된 내용에 따라 공정을 실행하기 위하여 생산에 필요한 가공요소들을 배치하는 역할을 한다. 특히 SFCS는 특정 공정 라우팅의 선택이나 자원할당, 작업물의 일정계획, 가공명령어 다운로드, 가공의 진척상황 모니터링, 고장 감지 및 진단, 제조시스템의 상태 보고서 작성 등을 행할 수 있어야 한다. 이렇게 다양한 컨트롤 동작을 일괄적으로 통합한다는 것은 쉽지 않다. 이러한 문제를 해결하기 위해 정형화된 세 가지 분류기법이 주로 사용되는데, 계층적 구조, 수평적 구조, 복합적 구조 등이 있다 (그림 1).



[그림 1] SFCS의 분류

계층적 SFCS는 National Institute of Standards and Technology (NIST) 에서 개발된 5 층 컨트롤 구조의 낮은 세 층에 근거를 둔다 ([2]). 계층적 컨트롤 구조를 구성하기 위해서, 장비 컨트롤러는 컨트롤러를 필요로 하는 물리적 장치와 연결되어 구성되어 있어야만 한다. 일단 각 장비 컨트롤러가 연결되어지면 장비집단을 배치할 수 있는 워크스테이션 컨트롤러가 필요하다. 워크스테이션 컨트롤러를 생성하기 위하여 필요한 장비컨트롤러 그룹은 제품을 생산하기 위하여 필요한 장비간 상호작용의 양과 공장의 레이아웃에 의존한다 ([4]).

수평적 SFCS는 계층적 컨트롤 구조에 존재하는 마스터/슬라브 관계를 없애고, 통신을 통하여 시스템의 목적을 달성한다. 네트워크 통신과 분산 컴퓨팅 시스템의 발전은 계층적 컨트롤러의 발전과 밀접한 관계가 있다. 이러한 컨트롤 구조 하에서 모든 종속 시스템은 자원에 대한 동등한 접근권을 가지며, 시스템 간에 상호 접근이 가능하다. 또한 작업이 서로 독립적이며, 전체 시스템의 프로토콜이 엄격하게 지켜진다. Lin & Silberg ([8]) 는 협상 베이스 협동 시스템 (negotiation-based cooperative system) 개념을 이용하여 shop floor control을 통합하였다. 그들은 컨트롤 프레임워크를 시장근거 모델로 제시하여 가격경쟁에 의한 협상으로써 시스템 목표를 달성하였다.

2.2 Execution function 기존 연구.

2.2.1 기능적 분류

대부분의 SFCS는 3가지 기능으로 구성된 시스템을 제안한다 ([2], [4]). Cho ([2]) 는 계층적 컨트롤 구조로 구성된 워크스테이션 컨트롤러를 제시하고, 기능적으로 3가지 (planning, scheduling, execution) 로 분류하고 이를 세부화 하였다 (표 1). 컨트롤러는 입력을 읽어 들이고, 주어진 입력과 시스템의 현재 상황을 참고하여 가공결정을 내리고, 결정되어진 내용을 실행에 옮기기 위하여 다른 컨트롤러와 정보를 교환하거나 통신을 하게 된다. 이러한 컨트롤러의 역할 중에서 결정을 내리는 기능은 planning 이나 scheduling 기능에 해당된다. 그리고 메시지를 받아들이거나 생성하는 역할은 execution function에 해당하게 된다. 계층적 SFCS 에서 execution function은 입력 메시지를 해석하고 고장을 감지하고 점검하며, 새로운 메시지를 다른 컨트롤러에 알리는 것이다 ([2]).

[표 1] SFCS의 기능 분류

	planning	scheduling	execution
shop	<ul style="list-style-type: none"> . getting and batching orders . batch routing . resource allocation 	<ul style="list-style-type: none"> . sequence batches . scheduling batches . deadlock resolution 	<ul style="list-style-type: none"> . monitoring workstation status . interfacing with factory level . communication . error detection and recovery
workstation	<ul style="list-style-type: none"> . splitting batches . resource allocation . part routing 	<ul style="list-style-type: none"> . sequencing parts . scheduling parts . deadlock resolution 	<ul style="list-style-type: none"> . monitoring equipment status . communication . error detection and recovery
equipment	<ul style="list-style-type: none"> . resource allocation . machining parameter optimization . tool path refinement 	<ul style="list-style-type: none"> . sequencing operations . merging NC codes 	<ul style="list-style-type: none"> . downloading NC code . monitoring device status . communication . synchronization . error detection and recovery

2.2.2 Execution Model 연구 고찰

산업체에서 널리 사용되고 있는 컨트롤러의

80% 이상은 단순한 모니터링 기능만을 수행하고 있다 ([6]). Thesis Group의 Auto-Cell, Arther Anderson의 Cell-Pace, Fast Tech Integration의 CELL works 등을 들 수 있다. 그러나 생산시스템의 execution function은 고장 감지 및 보정 또는 컨트롤 등의 기능도 추가되어야만 한다. execution function의 모델링에 가장 널리 사용되는 방법의 하나로는 페트리네트를 들 수 있다 ([3], [8], [9]). Merabet ([9])는 장비 동기화, 틀과 end-effect 변환 등의 기능을 담당하는 execution function을 페트리네트를 사용하여 모델링 하였다. Huang & Chang ([3])은 섹페트리네트를 이용하여 생산시스템에서 복수의 파트의 흐름을 효과적으로 컨트롤하였다([3]).

3. 기본 이론

3.1 민첩생산시스템

유연생산시스템은 자동화, 고객 요구 다양화 대응, 다품종 소량 생산을 목적으로 한 생산시스템이다. 또한 컴퓨터통합생산시스템은 유연생산시스템의 특징을 만족시키며 CAD나 CAPP, CAM을 통합한다. 자동설계, 자동공정계획, 자동비지니스 등을 통한 생산시스템의 전기능의 복합화를 꾀하였다. 이러한 생산시스템은 민첩성을 만족시키기에는 부족한 면이 많다. 민첩생산시스템의 주목적은 끊임없이 변하는 시장 환경에 효과적으로 대처하고, 예측이 어려운 소비자의 요구를 만족시키는데 있다. 이러한 조건을 만족시키며 고품질을 꾀하고, 확장되어 가는 시장에서 생존하는 생산시스템의 설계이다.

본 연구는 컴퓨터와 통신의 기능을 이용하여 생산시스템 내에서 가장 중요한 SFCS에서 민첩생산시스템을 구현하고자 한다. 이를 위해서는 기존에 주로 사용되고 있는 계층적 컨트롤 구조보다는 수평적 컨트롤 구조를 채택하였다. 수평적 컨트롤 구조는 시스템 내의 생산요소 상호간의 통신량이 많고, 생산지식의 분산화 등의 문제로 시스템 적용

이 회피 되었으나, 통신 기술의 발달과 컴퓨터의 고기능화로 이런 문제를 해결할 수 있게 되었기 때문이다.

3.2 Negotiation-based solution technique &

Cooperative systems

수평적 컨트롤 구조는 컨트롤 관리자가 존재하지 않는다. 즉, 생산을 위한 모든 결정은 시스템을 구축하는 요소들간의 상호 작용에 의하여 이루어지게 된다. 그러므로 시스템을 효율적으로 구축하기 위해서는 효율적인 상호작용 방법론이 필요하다. 본 연구에서는 컴퓨터 작업 할당 (computer task allocation) 과 밀접한 관계를 갖는 협상 베이스 협동작업 시스템의 방법론을 이용하였다. 협상 베이스 협동작업 시스템은 시스템을 이루는 agent간의 효과적인 상호작용을 목적으로 한다.

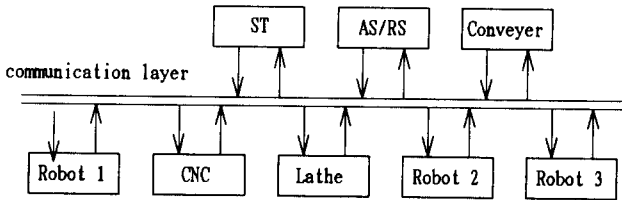
3.3 Petri-Nets

페트리네트는 여러 특성을 가진 시스템을 모델링할 수 있는 기법으로써 그래픽 기능과 수학적 해석능력을 특징으로 한다. 특히 동시적, 비동시적, 비확정적, 그리고 확률적인 특성을 가진 이산사건 동적시스템을 표현하고 해석하기 위해 유용하게 사용될 수 있다. 페트리네트의 그래픽 기능은 흐름도 (flow chart), 블록선도(block diagram) 및 네트워크와 같이 시스템을 그래픽으로 표현할 수 있음을 의미한다. 또한 토큰을 통하여 시스템의 동적인 상태를 나타낼 수 있다. 수학적 해석 기능은 상태방정식, 대수방정식 및 시스템의 지배방정식을 세울 수 있음을 의미한다. 이미 유연생산시스템의 제어와 모니터링 ([5])에서 실용성이 입증된 페트리네트를 이용하여 execution function을 모델링함으로써, 모델링 결과가 바로 가공지식으로 사용될 수 있게 하기 위함이다.

4. Execution Function of the system

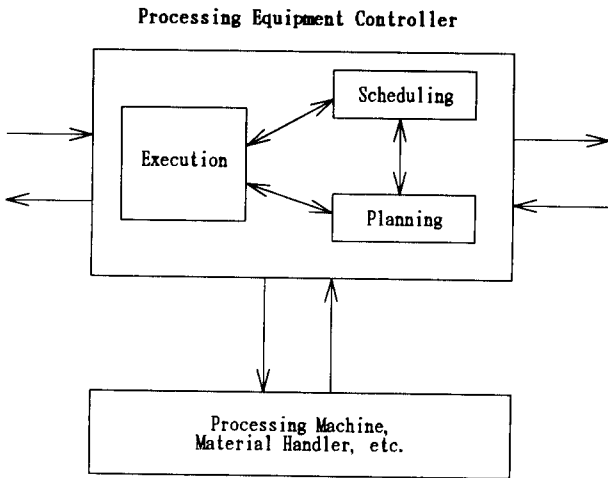
4.1 시스템 구조

본 연구에서 설계한 수평적 컨트롤 구조는 파트를 독립된 agent로 보지 않는다. 또한 이로 부터 발생하는 가공의 시작을 처리하는 문제를 ST(Support Tool)를 이용하여 해결하였다. 공정계획에서 작성된 파트의 가공정보는 ST에서 첫번 제작업이 선택되게 된다. 본 연구는 포항공과대학교 산업공학과 CIM Lab의 Postrol 시스템을 대상으로 하였다. Postrol은 2대의 이송로봇, 1대의 조립로봇, AS/RS, 컨베이어, 자동선반, 자동 밀링머신으로 구성되었다. 각 장비를 하나의 agent로 보면 시스템은 다음과 같은 구조를 하게 된다(그림 2).



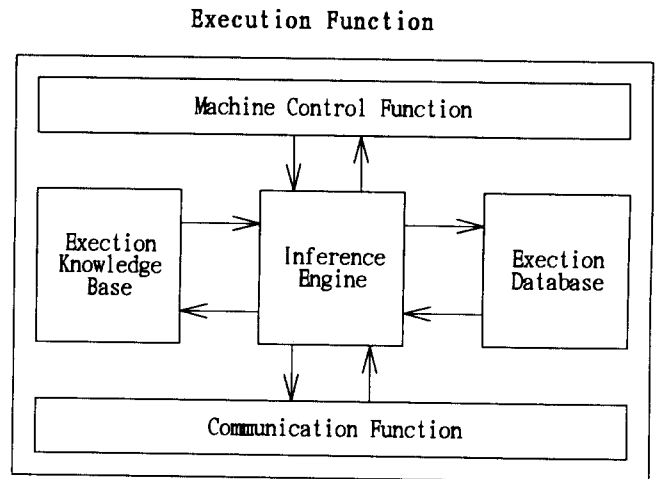
[그림 2] 컨트롤의 수평적 구조

위와 같은 컨트롤 구조에서 각각의 장비컨트롤러를 기능상으로 planning, scheduling, execution으로 분류하였다(그림 3).



[그림 3] 장비 컨트롤러의 기능적 분류

planning function과 scheduling function은 의사결정과 관련된 컨트롤러의 기능을 담당하게 된다. 그리고 execution function은 메시지 모니터링 및 해석, 실행베이스 의사결정, 협상, 고장 감지 및 진단, 메시지 전송 및 기계 코드 다운로드를 담당하게 된다. execution function에 메시지를 전달하는 객체는 의사결정기능 (일반계획, 일정계획), 실행 장비, 그리고 다른 컨트롤러이다. 이러한 객체로부터 메시지를 받아 실행베이스 의사결정을 하게 된다. 실행베이스 의사결정 과정에서 메시지를 해석한다. 대부분의 메시지는 실행 베이스 의사결정을 거친 후, 의사결정기능, 실행장비, 또는 다른 컨트롤러로 보내지게 된다. 이러한 과정을 처리하여야 하는 execution function은 지식베이스 시스템을 이용하였다(그림 4).



[그림 4] execution function의 지식베이스 시스템 구조

execution function 내에서 가장 중요한 부분은 추론기관과 실행지식을 저장하는 지식베이스이다. 본 연구에서는 시스템을 페트리네트를 이용하여 모델링한 결과를 지식베이스 구축에 이용될 수 있게 하였다. 추론기관은 지식베이스 내의 페트리네트 지식을 점화하는 페트리네트 점화 기관으로 구축되

었다 ([6]).

4.2 Messages and their description

execution function에 입력되는 메시지는 메시지의 출처에 따라 다음의 세가지로 분류될 수 있다.

- . 의사결정기능 메시지
- . 장비가공 메시지
- . 컨트롤 메시지

execution function으로 부터 출력되는 메시지 또한 목적지에 의하여 세가지로 분류된다.

- . 의사결정기능 메시지
- . 장비가공 메시지
- . 컨트롤 메시지

메시지의 입력부와 출력부를 기준으로 하여 메시지를 세부적으로 분류하였다 (표 2).

[표 2] 메시지의 분류

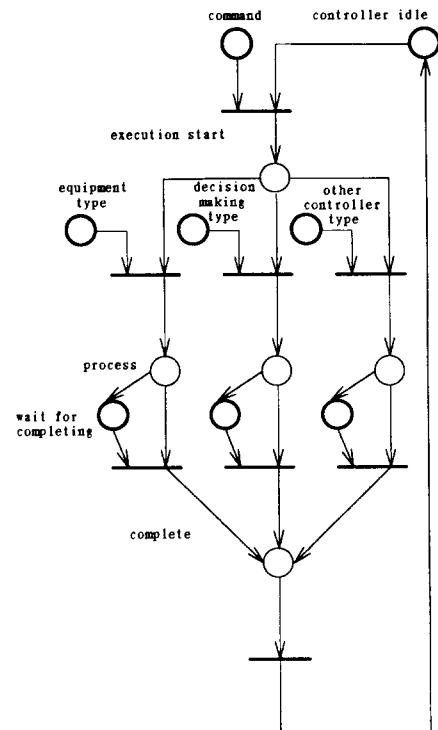
수신 송신	Decision Making	Equipment	Other Controller
Decision Making		. equipment control data . equipment status query	. processing start command . controller status query
Equipment	. machine status info. . processing status . controller error		. machine status info. . processing status . controller error
Other Controller	. part information . controller error	. equipment status query . equipment control data	

메시지의 분석은 컨트롤과 정보를 구분해야 한다 ([1]). 예를 들면 의사결정기능에서 실행장비에 보내는 move 메시지는 컨트롤메시지이다. 그리고 가공장비에서 의사결정에 move작업 완료 메시지는 보내면 정보 메시지라고 볼 수 있다. 컨트롤 메시지

와 정보 메시지는 분류되어 처리된다.

4.3. Petri-nets model

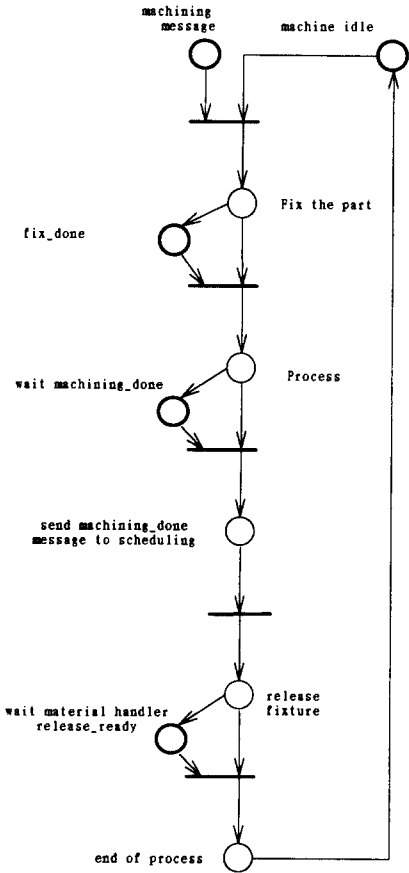
execution function은 페트리네트를 이용하여 모델을 구축하였다. 주페트리네트는 [그림 5] 와 같다. execution function이 idle한 상태에서 메시지가 입력되면 추론기관에 의해 주페트리네트가 점화된다. 입력된 메시지는 문장분석을 통하여 명령어의 목적지에 따라 equipment type, decision making type, other controller type으로 분류되어진다. 각 메시지를 처리할 수 있는 상태일 때까지 대기 후에 메시지를 처리한다. 메시지의 처리를 위해서 process 단계에서 부속페트리네트를 지식베이스로 부터 로드하여 이를 점화하는 과정을 거친다. 이런 과정을 반복하여 부속페트리네트가 종료되면 그 값이 주페트리네트에 전달된다. 메시지의 종료에 따라, 완료신호 대기후에 작업을 완료한다.



[그림 5] execution function의 주페트리네트

주페트리네트의 점화과정에서 부속페트리네

트를 다시 점화하게 되는데 다음은 CNC에 관련된 부속페트리네트이다. Postorol을 구성하는 CNC 장비이다. 먼저 파트를 고정시키기 위하여 이송로봇과 협상을 한다. 고정이 완료되면 가공을 시작한다. 가공종료 후에 가공종료 메시지를 의사결정기능에 전송한다. 그리고 파트를 언로드하기 위한 협상을 대기한다.



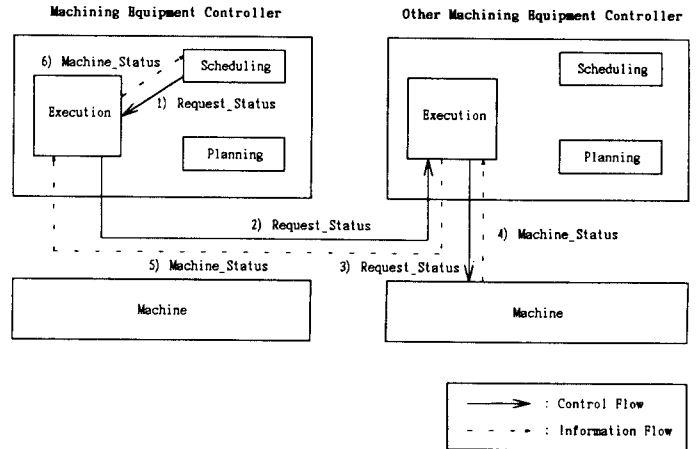
[그림 6] CNC execution function의 페트리네트 모델

4.4 execution function의 메시지 흐름의 예

이렇게 모델링되어진 execution function은 다양한 실행 베이스 의사결정을 하게 된다. 두가지의 예를 들어본다.

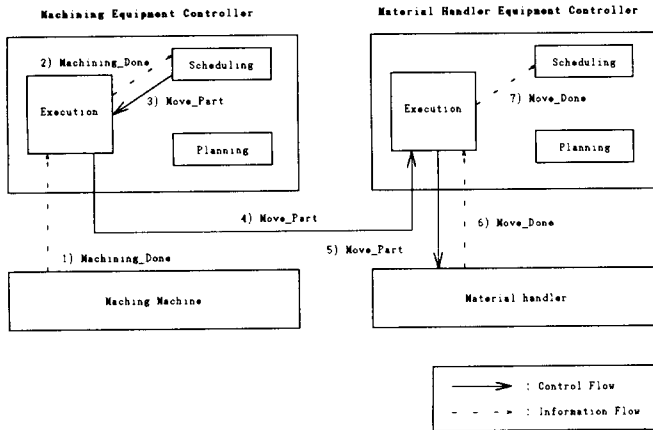
planning function이나 scheduling function은 다른 장비의 상태를 알아내기 위해 다른 머신과의 협상을 원하게 된다. 협상 베이스 해답을 얻기 위한 scheduling function의 명령을 수행하는 경우 파트를 가공중인 agent를 중심으로 협상이 이루어지게 된

다. 다음은 다른 agent의 상태정보를 원하는 상황이다. 주체가 되는 agent의 일정계획에서 장비상태 알아내기위한 메시지를 보내면, execution function이 다른 agent의 execution function과 컨트롤 메시지를와 정보 메시지를 교환하며 정보를 얻어내게 된다. 그리고 마지막으로 이를 scheduling function에 전달하게 된다.



[그림 7] 장비상태를 알아내기 위한 협상의 메시지 흐름

둘째는 이렇게 협상이 끝나고 다음 가공을 할 장비가 결정되어 있는 상태에서, 다음 agent에서 작업을 수행하기 위한 컨트롤 메시지와 정보 메시지의 흐름이다. 가공장비로부터 작업종료신호를 받은 execution function이 scheduling function에 이를 전달하면, execution function은 다음 작업을 하게 될 장비에 작업 메시지를 전송한다. 파트 관련 메시지 전송과 협상 메시지 전송 과정의 메시지 전송을 제외시키며 다음과 같은 메시지의 흐름이 존재하게 된다 ([그림 8]).



[그림 8] 파트이동을 위한 협상시 메시지 흐름

5. 결론

본 연구에서는 민첩생산시스템의 주요변수인 SFCS를 수평적 컨트롤 구조를 이용하여 구성하였다. 이를 기능상으로 컨트롤을 담당하는 execution function과 의사결정을 담당하는 planning, scheduling function으로 구분하고, 세가지 기능 중 execution function에 대해 협상 베이스 협동작업 해법을 이용하여 모델을 제시하였다. 모델링은 페트리네트를 이용하여 모델링의 결과가 바로 시스템에 적용될 수 있게 하였다.

[Reference]

- [1] Barkmeyer, E. J., "Some interactions of information and control in integrated automation systems," NATO ASI Series, Vol. F53, *Advanced Information Technologies for Industrial Material Flow Systems*, Springer-Verlag, New York, New York (1989).
- [2] Cho, H., *An Intelligent Workstation Controller For Computer Integrated Manufacturing*, Ph.D. Dissertation, Texas A&M (1993).
- [3] Huang, H. P. and Chang, P. C., "Specification, modeling and control of a flexible manufacturing cell," *International Journal of Production Research* 30, 11, 2515-2543 (1992).
- [4] Jones, A. T. and Saleh, A., "A multi-level/multi-layer architecture for intelligent shop floor control," *International Journal of Computer Integrated Manufacturing* 3, 60-70 (1990).
- [5] Kim, G., *A Study on the Control and Monitoring of FMS based on Petri Nets*, MS. Theses, Pohang University of Science and Technology (1993).
- [6] Larin, D. J., "Cell control: what we have, what we will be need," *Manufacturing Engineering* January, 41-48 (1989).
- [7] Lee, K., *Formal Integration of Process Planning and Shop Floor Control for CIM*, Ph.D. Dissertation, Pohang University of Science and Technology (1994).
- [8] Lin, G., Y-J., and Solberg, J. J., "Integrated shop floor control using autonomous agents," *IIE Transactions* 24, 3, 57-71 (1992).
- [9] Merabet, A. A., "Synchronization of operations in a flexible manufacturing cell: The petri-net approach," *Journal of Manufacturing Systems* 5, 3, 161-169 (1986).
- [10] Naylor, A. W. and Volz, R. A., "Design of integrated manufacturing system control software," *IEEE Transactions on Systems, Man, and Cybernetics* SMC-17, 6, 881-897 (1987).
- [11] O'Grady, P. J. and Menon, U., "A hierarchy of intelligent scheduling and control for automated manufacturing systems," *Proceeding of the symposium on real time optimization in automated manufacturing facilities*, National Bureau of Standards, Gaithersburg, Maryland, (January 15-29 1986).