

# 협동 생산 하에서의 조립 시스템 컨트롤 (Assembly System Control under CSCW-based Shop Floor Control)

손 경준, 조 현보, 정 무영

포항공과대학교 산업공학과

## ABSTRACT

이제까지의 생산 시스템 컨트롤은 여러 개의 계층을 통해 상위 컨트롤러가 하위의 컨트롤러를 관리하는 계층적/감독(hierarchical/supervisory)컨트롤 방식이 일반적이었다. 그러나 생산 시스템의 규모가 증가하고, 이를 운용하는 컨트롤 소프트웨어가 복잡해짐에 따라 전통적인 계층적/감독 컨트롤 방식은 몇 가지의 문제점을 가지게 되었다. 첫째, 계층간의 관리 구조가 복잡해짐에 따라 현장의 생산 정보가 의사결정을 하는 상위의 시스템에 즉시 전달되지 못한다. 이는 실시간 의사결정이 필요한 많은 컨트롤 문제를 야기시킨다. 둘째, 생산 시스템의 레이아웃 및 운영 방식에 변화가 생겼을 때 컨트롤 소프트웨어를 신속히 수정하기가 어렵다. 이러한 문제들을 해결하기 위한 방법으로 수평적/협동(heterarchical/cooperative) 컨트롤 방식이 제시된다. 이는 신속한 의사결정과 변화에 따른 수정이 용이하도록 가공, 조립, 물류 등 생산 시스템의 여러 단위 요소의 자율적인 컨트롤러가 계층이 없이, 서로 동등한 입장에서 협동을 통해 컨트롤이 이루어진다. 본 연구에서는 수평적/협동 컨트롤 방식하에서 조립 시스템이 가져야 할 기능과 컨트롤 방법을 제시한다. 아울러, 조립 시스템 컨트롤러가 다른 컨트롤러들과 교환하는 정보 및 컨트롤 대상을 규명한다.

**Keywords :** shop floor control system, heterarchical control, cooperative control, negotiation

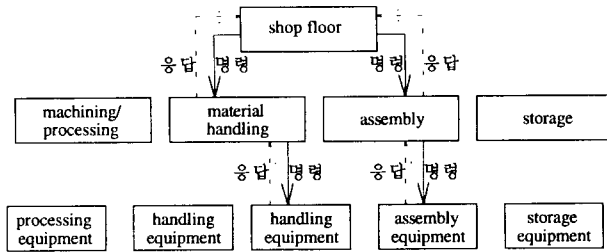
### 1. 서론

생산 시스템의 변화를 요구하는 시대적 상황에 따라 Shop Floor Control System(SFCS)은 그 구조와 기본 운용 방법이 변화되어 왔다. 생산의 유연성을 강조하는 FMS 나 생산과 정보의 통합을 요구하는 CIM 등은 이러한 변화 요구를 수용하기 위한 SFCS의 한 모델로 볼 수 있다. SFCS를 구조적인 측면에서 보면 컨트롤 범위에 따라 level을 둔 계층적

(hierarchical) 컨트롤 구조와 shop floor내에 주/종의 관계가 없는 수평적(heterarchical) 컨트롤 구조로 구분할 수 있다. 계층적 구조는 shop floor내의 컨트롤 문제들을 개념적으로 단순화 시켜 생각하기 위해서, 컨트롤 시스템을 여러 계층(level)으로 나누어 계층간의 주/종(supervisory/subordinate)관계에 의해 컨트롤 시스템이 운영되는 방법이다[1][2]. 상위 계층의 감독자(supervisor)는 하위 계층의 subordinate에

명령을 내리고 그 수행 결과를 하위의 기기 또는 컨트롤러로부터 받아들인다[그림 1]. 또한 감독자(supervisor)는 하위 계층의 기기들간의 상호 동작(interaction)을 조정하며 자신보다 하위의 시스템을 관리하는 역할을 맡고 있다. 현재도 이러한 계층적 컨트롤 구조에 관한 연구가 많이 이루어지고 있는데, 이는 계층적 컨트롤 구조의 이용이 복잡한 컨트롤 문제를 분할하여 해결하는 일반적인 방법이기 때문이다.

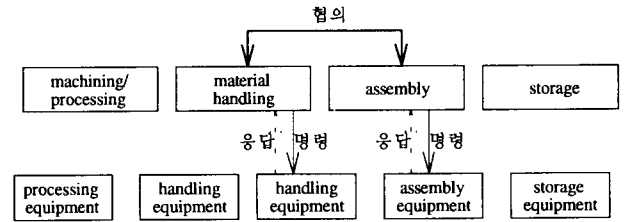
그러나, 계층적 컨트롤 시스템은 몇 가지 문제점을 가진다. 첫째, [그림 1]에서 보는 바와 같이 조립기기가 물류 설비로부터 가공물을 받아 작업을 하기 위해서는 상위의 컨트롤러로부터 명령을 받아야 하며, 그 결과를 다시 알려 주어야 한다. 다시 말해 실제의 작업과는 상관없는 부가적인 명령/응답의 전송이 이루어져야 한다. 둘째, 컨트롤러 중 하나가 장애를 일으키면 전체 시스템에 영향을 미치며, 그것이 상위의 컨트롤러일 경우에는 더욱 치명적이다. 그 밖에도 기기나 컨트롤러의 추가나 삭제와 같은 설비 배치의 변화에 신속하게 대응하기 어렵다.



[그림 1] 계층적(hierarchical) SFCS

heterarchical 컨트롤 구조는 계층적 컨트롤 구조의 이러한 문제점들을 해결해 보고자 하는 방법의 하나로 이는 컨트롤러들간의 계층을 없애고, 각 컨트롤러들이 동등한 입장에서 shop floor 를 운용해 나간다[3]. 즉, shop floor control에 있어서의 의사결정과 작업 지시를 컨트롤러간의 협의(negotiation)에 의해 능동적으로 수행한다[그림 2]. 이와 같이 heterarchical 컨트롤 구조가 계층적 컨트롤 구조의

문제점들을 해결할 수 있는 개념을 포함하고는 있으나 생산 정보의 분산 문제, 실시간 의사결정 문제, 그리고 컨트롤러들간의 협의에 따른 통신량 문제 등으로 실 시스템에의 적용은 거의 이루어지지 않았다.



[그림 2] 수평적(heterarchical) SFCS

컴퓨팅 환경의 고성능화와 대용량 전송이 가능한 통신 기술의 발달로 CSCW(Computer Supported Cooperative Work) 개념이 정보 시스템이나 통신 시스템에서 이용되면서, 생산 시스템 컨트롤도 그 응용 분야의 하나가 되었다. Shop floor control에 있어서 CSCW 개념의 도입은 heterarchical control 구조의 실현에 문제가 되었던, 실시간 의사결정 문제와 컨트롤러들간의 협의에 따른 통신량 문제 등을 해결할 수 있게 함으로써 수평적(heterarchical)/협동(cooperative) SFCS 구축의 가능성을 제시하고 있다.

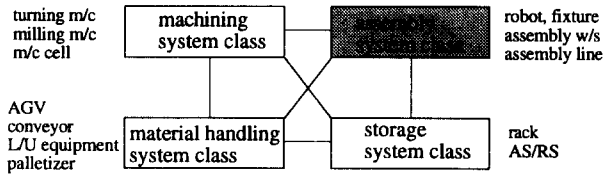
본 연구에서는 shop floor의 구성요소를 가공, 조립, 물류, 저장의 네 시스템으로 구분하고 각 시스템의 컨트롤러들이 어떠한 기능들을 가지는지 살펴본다. 특히, 조립 시스템은 어떠한 planning/scheduling 기능을 가지는지 고찰해 본다. 아울러 CSCW 환경하에서 각 컨트롤러들이 컨트롤 기능을 수행하기 위해 어떠한 협의 절차를 거치는지 알아보고자 한다.

## 2. CSCW 환경에서의 컨트롤 기능

### 2.1 shop floor의 구성

일반적으로 shop floor 는 [그림 3]에서 보는 바와 같이 가공 시스템, 조립 시스템, 물류 시스템, 저

장 시스템으로 구분할 수 있다. 각 시스템은 하나 또는 그 이상의 컨트롤 component를 가진다. 여기서 component란 하나의 컨트롤러에 의해 컨트롤 되는 기기들의 집합을 말한다. 예를 들어 가공 시스템 내의 processing component는 여러 대의 가공 기계로 구성되며, 경우에 따라서는 한대의 기계로만 구성될 수도 있다. 즉 가공 시스템의 규모가 크거나 컨트롤이 복잡할 경우에는 몇 개의 component로 가공 시스템을 분리할 수도 있다. 이런 경우에 분리된 component 각각에 컨트롤러가 붙게 된다. 조립 시스템, 물류 시스템, 저장 시스템도 마찬가지로 component가 구성 된다. CSCW 환경에서는 [그림 3]과 같이 full mesh 형태로 shop floor가 구성된다. 이는 각 컨트롤러들이 동등한 입장에서 협의를 통해 shop을 운영해 나가기 위한 것이다. 예를 들어 조립시스템은 가공 시스템이나 물류 시스템과 직접 협의를 할 수 있다. [그림 4]에는 네 가지의 시스템으로 구성된 shop floor의 예를 나타내었다.

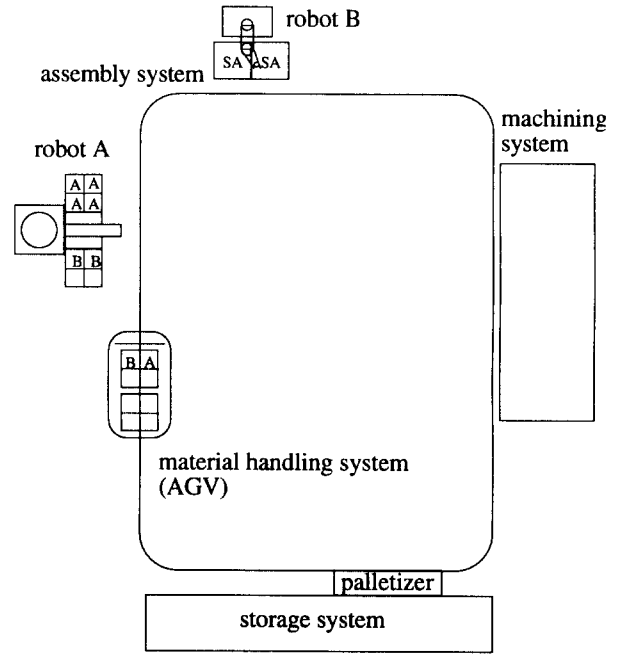


[그림 3] CSCW환경에서 shop floor의 구성요소

예로 든 shop floor는 두 대의 로봇으로 구성된 조립 시스템과 가공 시스템, 저장 시스템, 그리고 AGV로 구성된 물류 시스템으로 구성되었다. 조립 시스템의 입장에서 보면 나머지 세 개의 시스템은 조립 작업을 수행하기 위해 협의의 대상이 된다. 즉 조립에 필요한 부품을 저장 시스템이나 가공 시스템에 요구하고 부품 및 완성품의 이동을 위해서 물류 시스템 과도 협의를 하게 된다.

## 2.2 shop floor의 컨트롤 기능(control function)

shop floor내의 컨트롤 기능(control function)은



[그림 4] Shop floor의 예

planning기능, scheduling기능, execution기능의 세 가지로 구분할 수 있다. 각 기능이 수행해야 할 업무(task), 즉 의사결정(decision-making)과 실행(execution)의 범위를 얼마나 일반화 하느냐에 따라 컨트롤 기능의 수는 여러 가지로 분류 할 수 있다. Planning 과 control 의 두 가지 기능으로만 나눌 수 있으며, planning, scheduling, control 의 세 가지로 나누기도 한다[4]. 여기에 monitoring 기능을 추가하여 네 개 또는 그 이상의 기능으로 구분 하기도 한다 [5].

본 연구에서는 SFCS을 계층적 구조로 표현하고 특히 workstation level에서 SFCS의 기능을 planning, scheduling, execution의 세 가지로 구분 한 연구[6]를 바탕으로 하였다. 이에 따르면, SFCS의 planning 기능은 off-line의 공정 계획 시스템에서 생성된 feature graph를 입력받아 현재 shop floor의 상태를 고려하여 가장 바람직한 operation sequence graph를 생성하는 것이다. Scheduling 기능에서는 하나의 부품을 가공하기 위해, planning 기능으로부터 각 부품에 대한 operation sequence graph를 입력받아

execution 기능을 통해 각종 기기에 [표 1]과 같은 명령(message)을 내린다.

[표 1] scheduling 기능의 결과

system class	outputs
machining system	refixture(part2, machine1)
material handling system	transfer(part1, machine1, robot) move(part1, location1, location2)
assembly system	peg_in_hall(part1, part2)
storage system	retrieve(part1, storage1)

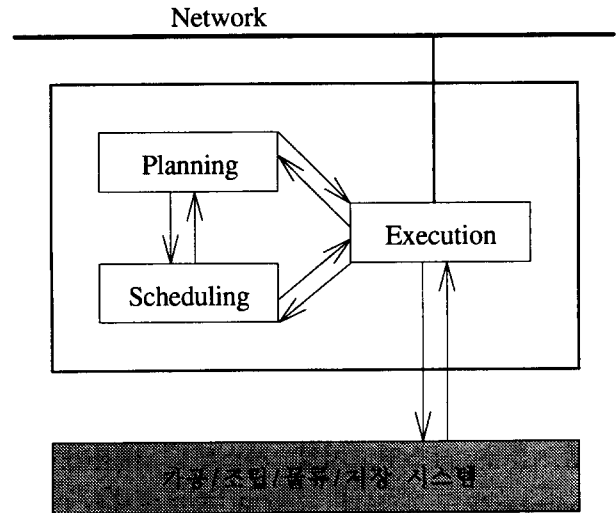
[표 1]에 나타난 바와 같이 scheduling 기능에 의해 생성된 명령은 메시지의 형태로 각 기계에 전송된다. 전송되는 명령의 집합은 단순히 operation의 순서만을 나타내는 것이 아니다. 명령의 전송은 컨트롤러간의 협의(negotiation)에 의해 실시간으로 변하는 shop floor의 상태를 고려하여, 적절한 시간에 이루어져야 한다.

Execution 기능에서는 scheduling 기능에서 넘어온 명령들을 각 동작기계에 적합한 통신 프로토콜로 변환시켜 해당하는 동작을 수행하게 한다. Execution 기능은 통신 네트워크와 밀접히 연관되어 있으며, 컨트롤러들간의 협의 창구 역할을 한다. 또한 해당 기기들의 상태를 모니터링 하여 scheduling, 또는 planning에서 참조할 수 있도록 한다. [그림 5]에 세 가지 기능으로 구성된 컨트롤러를 나타내었다.

### 2.3 조립 시스템의 컨트롤 기능

Shop floor내에서 조립 시스템이 하는 일은 저장 시스템과 물류 시스템에 필요로 하는 부품들을 요청하여 완제품을 만들어 내는 것이다[7]. 따라서 조립 시스템의 컨트롤러는 이 두 시스템 컨트롤러와 긴밀한 상호작용(interaction)에 의해 작업을 수행해 나가야 한다. 경우에 따라, 필요로 하는 부품을 가공을 통해서 얻어야 한다면, 조립 시스템 컨트롤러

는 가공 시스템 컨트롤러와도 협의를 해야 할 것



[그림 5] 컨트롤러의 세 가지 기능

이다. 조립 시스템 컨트롤러가 가져야 할 planning 기능에는 다음과 같은 것들이 있다.

- 각조립품에 대한 operation graph 생성  
Off-line에서의 조립계획은 assembly plan을 생성하게 되는데, 그 표현 방식은 여러가지가 있다[7]. SFCFS의 입력으로 들어오는 assembly plan은 가능한 조립경로를 모두 포함하는 것이 바람직 한데, 이는 shop floor의 상황에 따라 그 순서를 달리 할 수 있는 유연성을 가질 수 있기 때문이다. 이러한 표현방식은 AND/OR graph의 형태가 가장 일반적인데, planning기능은 이를 scheduling기능에 넘겨줄 수 있는 operation graph를 생성한다.
- part mix에 따른 equipment 프로그램 선택  
[그림 4]의 shop floor에서 조립로봇 robotA는 pallet에 실려 온 부품A와 B를 조립하는 기능을 한다. 이때 pallet에 담긴 부품이 어떻게 적재 되었는가에 따라 로봇의 동작은 달라야 한다.
- shop floor내의 lot sizing 결정  
생산의 효율을 위해 shop floor내에는 lot의 결정이 필요한 경우가 있다. 예를 들어, [그림 4]

에서 AGV가 한꺼번에 두개의 pallet을이송할 수 있고 하나의 pallet에는 4개까지의 부품을 적재할 수 있다고 가정해 보자. 조립시스템에서 8개의 부품을 요구하는데 재고는 5개이고 3개는 가공시스템에서 가공중이라 한다면, 하나의 pallet만을 먼저 이송해야 할지, 가공이 끝난 후 2개의 pallet을 한꺼번에 이송해야 할 지를 결정해야 한다.

Scheduling 기능에서는 다음의 역할을 갖는다.

- cycle time과 makespan의 계산  
여러가지 scheduling 문제에는 시간의 제약이 따른다. 예를 들어 조립 시스템에서 잔여 작업 시간이 10분 이고, 새로운 부품을 5분이 걸리는 가공을 통해 공급 받아야 한다면 늦어도 5분 후에는 가공 시스템과 협의를 해야 한다.
- 협의 시간(negotiation timing) 결정
- 예기치 못한 shop의 상태변화로 현재의 schedule을 맞추지 못할 경우 planning 기능에 replanning 요구한다.

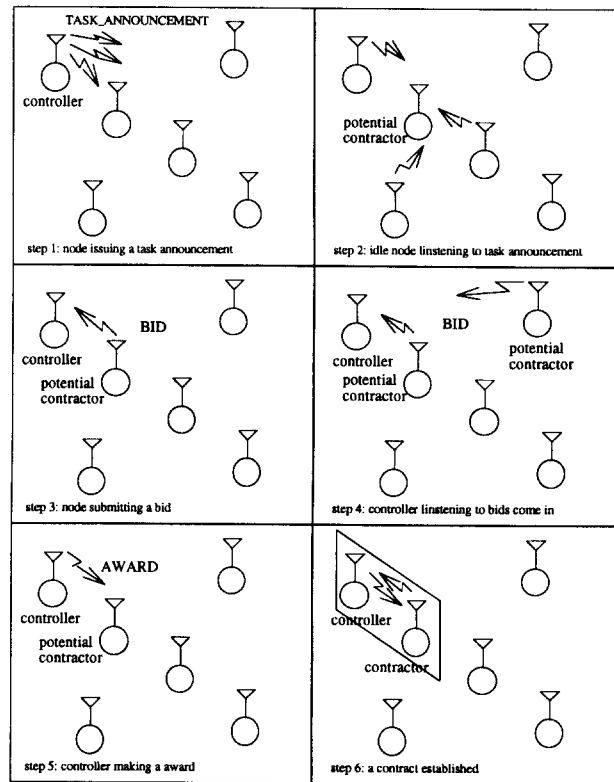
### 3. 협의 중심의(negotiation-based) control

#### 3.1 협의 메시지 형식

CSCW 환경하의 컨트롤러 들은 상위 level 감독 (supervisor)의 개입 없이 능동적으로 서로간의 실시간 협의(real-time negotiation)에 의해 planning, scheduling, execution의 기능을 수행해 나간다. 협의의 주체가 부품-자원 이 되는 경우도 있으나[8], 본 연구에서는 부품은 수동적인 요소로 보고 자원-자원간의 경우만을 그 대상으로 한다. 여기서 자원이란 각 시스템의 컨트롤러를 말한다.

컨트롤러간의 협의절차는 contract net protocol을 바탕으로 하여 이루어지는데[9], 그 과정은 [그림 6]과 같다. [그림 6]에 나타난 협의 과정은 contract net 의 기본 protocol로 best bid를 제시하는 노드와의 contract 가 이루어지는 과정을 나타낸다. 각 노드는 SFCS의 컨트롤러에 대응되며 bid는 task를 수행하는데 걸리는 시간 등으로 contract 결정의 기준이 된

다. Contract net protocol의 메시지 형식을 바탕으로 SFCS에서 나타날 수 있는 메시지의 형식을 [표 2]에 나타 내었다.



[그림 6] Contract net protocol

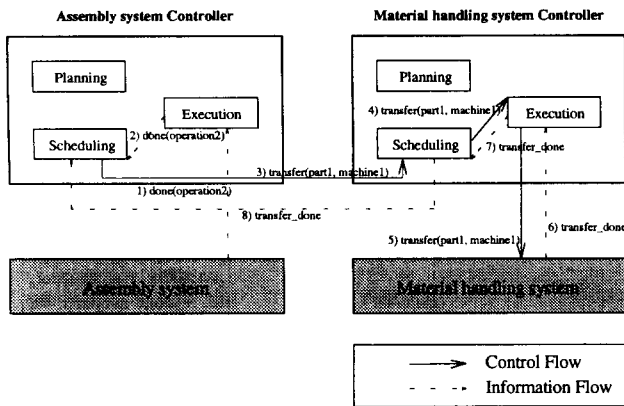
[표 2] 협의메시지 형식

	message type	example
request	QUERY	assembly system query to storage system the number of available pallets
	REQUEST	claim AGV to deliver finished product
	AWARD	accept the best bid
	TASK_ANNOUNCEMENT	broadcast needed task and start bidding procedure
response	ANSWER	reply the number of available pallets
	YES/NO_WAY	AGV is busy now
	BID	submit bid
notice	NOTICE	machine 1 breakdown

Request는 어느 컨트롤러가 다른 컨트롤러(들)에게 어떤 task나 정보의 요구를 나타내며, response는 이에 대한 응답, 또는 bidding 정보를 나타낸다. Notice는 하나의 컨트롤러가 다른 컨트롤러(들)에게 요구사항이 없는 메시지의 전달로 기계의 고장통지 등을 들 수 있다.

### 3.2 컨트롤 명령 흐름

CSCW 환경의 컨트롤 시스템은 각 컨트롤러의 scheduling기능에서 컨트롤러간의 협의와 명령전송을 담당한다. 협의는 컨트롤러의 scheduling 기능 간에 의사결정을 하는 것이며, 명령전송은 수행하고자 할 명령 메시지를 execution기능에 내리는 것이다. Execution 기능은 scheduling기능에서 내려온 명령 메시지를 수행하고 그 결과를 schedulin기능에 보고한다. 예를 들어 조립 시스템에서 조립 작업이 끝난 공작물을 이동시키기 위해 물류 시스템의 컨트롤러에 이송을 요청하고자 할 때의 메시지 전송 절차를 [그림 7]에 나타 내었다.



[ 그림 7 ] 명령전송 절차의 예

[그림 7]에서 보는 바와 같이 조립 시스템의 작업이 끝났음을 알리는 메시지가 execution 기능을 통해서 들어오면 조립 시스템 컨트롤러의 scheduling 기능은 다음 작업을 수행하기 전에 물류 시스템의 컨트롤러에 공작물의 이송을 요청한다. 물류 시스템 컨

트롤러의 scheduling 기능은 이러한 요청을 받아 execution 기능에 이송 동작 명령을 내리고 그 결과를 scheduling 기능에 보고한다.

### 4. 결론

CSCW-based 컨트롤 모델은 shop floor controller의 개발 시간을 단축시킬 수 있으며, shop floor의 배치 및 기능 변경에 신속하게 대응할 수 있다. 따라서 생산 환경의 급격한 변화에 민첩하게(agile) 대응할 수 있다. 본 연구에서는 CSCW환경하의 shop floor가 어떻게 구성되어야 하며, 특히 조립 시스템의 planning과 scheduling 기능은 어떠한 역할을 담당해야 하는가를 제시하였다. 그러나 이러한 역할들이 구체적으로 어떻게 구현되어야 하는지는 앞으로의 연구가 필요하다. 또한, Off-line의 조립 계획 시스템과 조립 시스템의 planning기능간의 관계에 대한 연구가 이루어져 실 시스템에 적용 가능한 컨트롤러의 개발이 요구된다.

### 참고문헌

- [1] O'Grady, P. J. and Menon, U., "A hierarchy of intelligent scheduling and control for automated manufacturing systems," *Proceedings of the symposium on real time optimization in automated manufacturing facilities*, National Bureau of Standards, Gaithersburg, Maryland, (1986).
- [2] Joshi, S., Wysk, R. A., Jones, A. T., "A scaleable architecture for CIM shop floor control," *Proceedings of CIMCON '90*, NIST, Gaithersburg, Maryland, 21-33 (1990).
- [3] James, J. Ting, "A Cooperative shop-floor control model for Computer-Integrated Manufacturing," *Proceedings of CIMCON '90*, 446-465 (1990).
- [4] Stecke, K. E., "Design, planning, scheduling, and control problems of flexible manufacturing systems," *Annals of Operations Research* 3, 3-12 (1985).

- [5] David, W., Jones, A., and Saleh, A., "Generic architecture for intelligent control systems," *Computer Integrated Manufacturing Systems* 5, 1, 105-113 (1992).
- [6] Cho, H., *An Intelligent Workstation Controller For Computer Integrated Manufacturing*, Ph.D. Dissertation, Texas A&M University (1993).
- [7] Delchambre, A., *Computer-aided Assembly Planning*, Chapman & Hall (1992).
- [8] Lin, G., Y-J., and Solberg, J. J., "Integrated shop floor control using autonomous agents," *IIE Transactions* 24, 3, 57-71 (1992).
- [9] David, R., and Smith, R. G., "Negotiation as a Metaphor for Distributed Problem Solving," *Artificial Intelligence* 20, 1, 63-109 (1983).