

FMC 반송용 로봇의 선전형 운영방법

윤정익¹ · 엄인섭¹ · 이홍철^{1†}

Proactive Operational Method for the Transfer Robot of FMC

Jung-Ik Yoon · In-Sup Um · Hong-Chul Lee

ABSTRACT

This paper shows the Applied Q-learning Algorithm which supports selecting the waiting position of a robot and the part serviced next in the Flexible Manufacturing Cell (FMC) that consists of one robot and various types of facilities. To verify the performance of the suggested algorithm, we design the general FMC made up of single transfer robot and multiple machines with a simulation method, and then compare the output with other control methods. As a result of the analysis, the algorithm we use improve the average processing time and total throughputs as well by increasing robot utilization, reversely, by decreasing robot waiting time. Furthermore, because of ease of use compared with other complex ways and its adoptability to real world, we expect that this method contribute to advance total FMC efficiency as well.

Key words : Q-learning, Simulation, Transfer robot, Flexible manufacturing cell

요 약

본 논문에서는 로봇 한 대를 중심으로 여러 설비들이 제어되는 Flexible Manufacturing Cell(FMC)에서 로봇의 대기위치 선정과 작업물 선택 의사결정을 위하여 Q-learning 알고리즘을 응용한 방법을 제시하였다. 이를 위해 하나의 로봇과 다수의 설비로 구성된 전형적인 가상의 FMC를 시물레이션으로 설계, 제안한 알고리즘을 적용하여 다른 대안과의 비교 분석을 실시하였다. 비교결과 적용된 알고리즘이 로봇가동률을 높여 평균 처리시간, 총 처리량 모두 향상시키는 결과를 가져왔다. 또한, 이 방법은 본 논문에서 제시한 정보뿐만 아니라 그 우선순위와 가중치에 따라 다른 정보들을 손쉽게 추가하여 적용할 수 있으므로 FMC의 생산성 향상에도 크게 기여할 것으로 기대된다.

주요어 : 큐러닝, 시물레이션, 반송로봇, 유연생산셀

1. 서 론

초기 공장자동화는 사용자가 대량생산된 제품 가운데 원하는 품목을 선택하는 소품종 대량생산 체제였다. 그러나 고객의 요구가 다양화되고 제조업체의 경쟁이 심화되면서 다양한 제품을 소량 생산하는 제조시스템의 확대를 통하여 제품의 사양이 달라져도 최소의 변경만으로 생산

이 가능하도록 생산원가를 낮추는 방식이 필요하게 되었다. 따라서 비교적 높은 생산성을 유지할 수 있는 자동생산시스템으로서 등장한 것이 유연생산시스템(Flexible Manufacturing System : FMS)이다³⁾. FMS는 부품가공 시스템을 가리키는 경우가 많은데, 부품가공 FMS는 머시닝 센터 등의 Machine, 가공 대상물의 Loader, Unloader, AGV (Automated Guided Vehicle), 자동창고(AS/RS), 제어용 컴퓨터 등의 하드웨어로 구성된다. 이러한 하드웨어는 생산계획 프로그램, 기계제어 프로그램 등에 의해 관리된다. FMS의 구성요소 중 Machine의 대수와 배치에 따라서 여러 가지 형식으로 분류 될 수 있는데, 그중 유연생산셀(Flexible Manufacturing Cell : FMC)은 작업물의 상차/loading), 하차(unloading)를 위한 한 대의 로봇과 자재 운반 작업을 위한 파레트(Pallet) 혹은 컨베이어(Conveyor)

* 이 연구에 참여한 연구자는 '2단계BK21사업'의 지원비를 받았음.

2008년 11월 18일 접수, 2008년 12월 13일 채택

¹⁾ 고려대학교 정보경영공학부

주 저 자 : 윤정익

교신저자 : 이홍철

E-mail; hclee@korea.ac.kr

와 하나 혹은 그 이상의 Machine들로 구성되어 있는 소규모 작업센터를 말한다. FMC는 한 대의 로봇을 중심(robot-centered cells)으로 시스템이 제어되기 때문에 대부분 원형, 직선형, U형 등의 Machine 배치 구조를 가지고 이에 따라 Machine간 작업물의 운반 방법이 결정된다³⁾.

FMC는 그룹 테크놀로지(Group Technology)를 적용하여 구성된 부품 패밀리(Part Family)에 속하는 부품들을 가공하는 기계셀의 기능을 가지기도 한다. 따라서 FMC는 때때로 GT 기계셀 이라고도 불리 운다³⁾. FMC내에는 다른 여러 종류의 작업물들이 동시에 가공되기 때문에 작업물의 스케줄링, 로봇, Machine의 특성에 따라 시스템의 성능이 결정된다. 그렇기 때문에 로봇의 작업물 선택과 대기위치는 시스템 퍼포먼스(performance)에 상당히 중요한 요소로 작용한다. 평가 척도로는 작업물의 총 처리 시간, 작업물의 평균 체류시간, 로봇의 가동률, 단위 시간당 생산량 등을 고려 할 수 있고 평가 척도들의 수행도에 영향을 주는 요인으로는 작업물의 남은 공정시간, 작업물의 다음 공정에 작업물 유무, 로봇의 현재 대기위치 등이 있다.

FMC로 작업물이 투입되면 현재 로봇의 작업수행이 자유롭다면 상차작업을 거쳐 작업물의 공정순서에 따라 해당 Machine으로 운반되고 정해진 공정을 통해 모든 작업을 마칠 때 까지 로봇에 의해 계속적으로 운반되어 진다. 로봇의 입장에서 보면 작업물 별로 공정순서와 작업 시간이 다르기 때문에 선 작업을 마치고 대기 상태가 되었을 때 대부분 마지막 작업이 이루어 진 장소에서 다음 신호가 올 때 까지 대기한다. 그리고 하나의 작업 완료 신호가 발생하면 그 작업을 수행하면 되지만 두 개 이상의 작업물에서 신호가 왔을 시 로봇은 어떤 작업물을 운반할지 결정해야 한다. 따라서 다음 작업물을 선택하는 방법에 대한 연구는 계속 되어져 왔다. Park(1992)³⁾은 선입선출 규칙, 최소 이동시간 우선 규칙, 최대 잔여 가공시간 우선 규칙, 최소 공정 순서 우선 규칙, 최대 공정 순서 우선규칙 5가지 대안을 비교하여 작업물의 공정순서가 가장 큰 번호로써 표현되는 작업물을 우선적으로 운반하는 최대 공정 순서 우선규칙을 최적의 대안으로 설명하였고, Savsar등(2008)¹⁸⁾은 두 machine간의 상황을 고려하여 수학적으로 확률적 모형(stochastic model)을 개발하였다. 또한 Sabuncuoglu 등(1989)¹⁵⁾, Henneke 등(1988)¹²⁾, Cogun 등(2006)¹⁰⁾은 로봇과 컨베이어의 스피드, machine의 배치등을 고려하여 확률적 모형과 시뮬레이션 모델을 개발하였다.

그러나 복잡한 시스템의 경우에는 수리적 모형으로 최적 해를 구하는 것이 어렵고 로봇이 선 작업 완료 후 다음 신호가 발생하지 않았을 때 로봇의 대기위치에 대한 연구가 필요한 상황이다. 또한 정해진 수행도 평가 요소들에 한에서만 연구들이 진행되어 왔기 때문에 현실에 적용하는데 어려움이 있다. 따라서 본 논문에서는 지능형 로봇의 주어진 환경에 적절한 행위를 학습할 수 있는 효과적인 방법으로 많이 활용되고 있는 Q-learning 알고리즘을 응용한 방법을 제시한다. Q-learning 알고리즘은 로봇의 학습효과를 통해 보상의 개념을 이용하여 주로 최적의 해를 얻는 목적으로 사용되지만 본 논문에서는 작업에 대한 우선순위를 결정하는 목적으로 사용하기 때문에 보상의 개념을 제외 하였다. 즉, 로봇 자신의 행동을 결정하는 방법으로 사용하는 것이 Q-learning 알고리즘이므로 과거 행위에 대한 정보를 갱신해야 하기 때문에 보상의 개념이 필요하다. 그러나 본 논문에서는 로봇 자신의 행위에 대한 정보를 기준으로 다루는 것이 아니라 목표물 탐색에 의의를 두기 때문에 보상의 개념을 적용하지 않았다.

본 논문에서는 로봇이 선 작업을 완료하고 대기 상태가 되었을 때 자신의 영역에 위치한 작업물들의 정보들을 이용하여 작업 완료 신호가 2개 이상 일 때에는 우선순위를 결정하여 수행하고 작업 완료 신호가 없는 경우 그 위치에서 작업 완료 신호가 발생할 때 까지 대기하지 않고 작업의 우선순위를 결정하여 먼저 행동할 수 있는 방법에 대하여 논한다. 그리고 제시한 방법을 시뮬레이션을 통하여 신호가 발생 하였을 때 반응하는 방법 등의 다른 대안들과 비교 검토하고 유효성을 검증한다.

본 논문의 구성은 다음과 같다. 2장에서는 반송로봇의 작업 우선순위 결정을 위해 사용된 Q-learning 알고리즘과 모의실험에 관한 이론적 배경을 설명하고 3장에서는 제안된 알고리즘을 설명한다. 4장에서는 사례연구(Case study)로서 제시한 방법의 모의실험 모델링에 관하여 설명하고 5장에서는 제안된 모델에 대한 실험 및 분석을 통하여 유효성을 검토한다. 마지막으로 6장에서는 결론과 향후 연구과제에 대해 설명한다.

2. 이론적 배경

2.1 Q-learning 알고리즘

최근 들어, 제어(Control), 계획(Planning), 상황 결정(Decision Making)등을 실시간으로 처리하고자 하는 지능형 시스템(Intelligent System)에 관한 연구가 많은 관심을 끌게 되었다¹²⁾. 로봇에 의해 제어되는 복잡한 시스템의

경우 다수의 경우에 대하여 관리하고 제어하는 것은 매우 어려운 문제이다. 강화학습은 이러한 목적을 달성하기 위해 개발된 학습 방법으로 환경에 대한 충분한 지식이 없어도 주어진 환경에 적절한 행위를 학습할 수 있는 효과적인 방법이다^[2]. 또한 강화학습은 에이전트로 하여금 주변 환경의 탐색을 통해 능동적으로 환경에 대한 행동을 결정하도록 한다. 보상값이 존재하는 어떤 불확실한 영역을 탐색하는 동안 에이전트는 연속적인 상태 공간을 따라 적절한 보상값을 전달함으로써, 임의의 상태에 대해 어떠한 행동을 취해야 할지를 학습하게 된다^[2]. 이러한 문제를 풀기위해 가장 널리 사용되는 것이 Q-learning 알고리즘이다. Q-learning은 미래의 행위에 대한 보답에 감쇠를 고려한 평가척도 Q값을 추정하여 주어진 현재 상태로 부터 최적의 행위를 찾아내는 알고리즘이다^[2]. Q-learning을 이용하여 연구한 사례들을 살펴보면 서상욱 등(2007)^[4]은 다수의 로봇이 존재하는 시스템에서 서로의 행동을 협조하는 방법을 위하여 사용하였고, 윤한얼 등(2005)^[5]은 다수 로봇의 목표물 탐색을 위하여 Area-Based Q-learning 알고리즘을 사용하였고, Marc Carreras 등(2007)^[9]은 로봇이 자신의 행동들을 Network를 이용하여 Real-time으로 정보를 교류하며 제어할 수 있는 Semi-Online Neural-Q-learning(SONQL) 알고리즘을 제시하였다.

Q-learning 알고리즘을 정리하면 표 1과 같이 나타낼 수 있다. 로봇이 의사결정을 하기 위하여 최초 현재 상태 s 값과 행동 a 값을 초기화 시킨다. 그 후 시행해야 하는 명령을 a 값에 입력한다. 로봇은 행동에 대한 결정을 위하여 현재 자신의 상태를 s 값에 입력 하고 기존에 알고 있는 Q값들 중 제일 큰 값을 선택하고 이에 따른 r 값을 보상 받는다. 최적의 행동을 얻어내면 Q 값을 갱신한다. 여기서 γ 는 Q - 함수값의 조정을 위한 계수(discount factor)로서 값이 커지는 것을 방지해 주는 역할을 한다. Q 값은 식 (1)에 따라 갱신된다.

$$\hat{Q}(s,a) \leftarrow r + \gamma \max_{a'} \hat{Q}(s,a') \quad (1)$$

2.2 모의실험(Simulation)

시뮬레이션은 시스템의 분석과 프로세스의 운영 설계를 위한 분석에 복합적으로 사용되는 분석 방법이다. 또한 최근의 컴퓨터 시스템의 발전으로 컴퓨터 시뮬레이션을 주로 이용하고 있다. 수학적 모델링의 경우 시스템의 전체를 설계하고 분석하는데 많은 어려움을 가지고 있기 때문에 최근에는 시뮬레이션 분석이 전체 시스템의 성과 측정 및 분석에 많이 활용되고 있다^[6]. 컴퓨터 시뮬레이션

표 1. Q-learning 알고리즘

<p>For each s, a initialize table entry $\hat{Q}(s,a)$ zero observe the current state s. Continue to infinity. - Select action a and execute. - Receive immediate reward r. - Observe new state s'. - Select action. - Update table entry for $\hat{Q}(s,a)$. - $s \leftarrow s'$</p>

은 로봇 제어분야 뿐만 아니라 다양한 분야에서 활용되고 있다. 예를 들어 자동창고, AGV, Conveyor등과 같은 설비들이 서로 연결되어 있는 통합 물류 시스템, 자동차 생산라인, LCD 생산라인 등 과 같이 분석하고자 하는 범위가 방대한 경우에 컴퓨터 시뮬레이션을 활용하고 있다.

로봇 제어를 위하여 컴퓨터 시뮬레이션을 활용한 사례들을 살펴보면 김재현 등(2007)^[2]은 Region-based Q-learning 알고리즘의 효율성을 기존의 Q-learning 알고리즘과의 비교를 위하여 사용하였고, 서상욱 등(2007)^[5]은 Polygon-based Q-learning 알고리즘을 군집 로봇의 두 가지 협조 행동에 대하여 모의실험을 수행하였다. 또한 Carreras 등(2007)^[9]은 로봇의 행동을 실시간으로 학습하도록 Semi-Online Neural-Q-learning(SONQL)의 유효성을 검증하기 위해 모의실험을 사용하였다.

3. 제안된 알고리즘

본 논문에서는 반송로봇의 작업물 선택 방법에 관하여 Q-learning 알고리즘을 응용 하였다. 표 1과 같이 설계된 시스템이 있다고 가정하면 한 로봇이 담당하고 있는 machine(MC)은 5개이다. MC 6은 다음 Zone의 반송 로봇에 의해 제어 되므로 제외된다.

먼저 작업물은 그림 1과 같이 화살표 순서대로 반송로봇에 의해서 운반되고 각각의 MC들은 특정 작업시간을 가지고 있다.

두 번째, 반송 로봇은 작업물의 운반을 위해 어떤 작업물을 선택 할지 결정해야 한다. 모든 MC에 작업물이 존재 하고 반송로봇의 마지막 행동은 MC 1에서 MC 2로 작업물을 운반했다고 가정 하자. 이때 반송 로봇은 대기 상태이고 현재 위치는 MC 2가 된다. 로봇의 사이클 타임을 감소시키기 위해서는 다른 MC에서 작업 완료 신호가

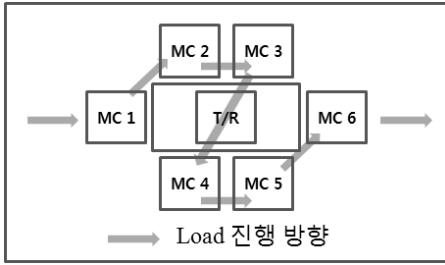


그림 1. 반송용 로봇 한 대가 담당하고 있는 작업장

표 2. 작업 우선순위 선택을 위한 정보

로봇 위치	factor a	factor b	factor c
MC 1	5	10	5
MC 2	10	6	0
MC 3	6	6	6
MC 4	6	4	7
MC 5	4	5	9

발생할 때 까지 대기하는 것 보다는 다음 작업을 미리 선택하여 해당 MC 앞에서 대기하는 것이 효과적이다. 따라서 현재 사용할 수 있는 정보들을 활용하여 우선순위를 결정하고 해당 MC로 이동하여 대기한다.

우선순위를 결정하기 위해서 표 2와 같이 각 작업물들의 현재 남은 작업시간(factor a), 각 작업물들의 다음 MC에 위치한 작업물의 남은 작업시간(factor b), 현재 로봇의 위치에서 각 MC들로 이동하는데 걸리는 시간(factor c)의 정보들을 활용한다.

표 2의 정보들을 이용하여 반송로봇은 다음 작업을 선택 하여야 하는데 먼저 로봇의 현재 위치를 파악(MC 1 = 1, MC 2 = 2...) 하여 s값을 결정하고(s = 2) 다음 행동에 대한 a값을 초기화 한다. 그 후 각 MC(i)별로 Q 값을 계산하는데 반송로봇의 경우 현재 위치에서 다른 MC로 이동해야 하는 경우 이동시간이 짧을수록 좋다. 그리고 작업물의 경우에는 현재 남은 공정시간이 적을수록, 다음 공정의 남은 공정시간이 적을수록 이동에 대한 요구가 빨리 다가 오기 때문에 각 factor(j)의 합이 작은 값을 취하는 것이 좋다. 따라서 식 (2)를 통하여 각 MC별 \hat{Q} 값을 계산하였다.

$$\hat{Q}(s', a') = \sum w_j f_{i_j} \quad (2)$$

각 factor별 중요도에 따라 표 3과 같이 가중치(ω)를 부여하였는데 반송로봇은 작업물의상태에 따라 작업 우

표 3. factor별 가중치

factor	a	b	c
ω_i	0.6	0.3	0.1
조건	$\omega_1 + \omega_2 + \omega_3 = 1$ $\omega_1, \omega_2, \omega_3 \geq 0$		

표 4. MC 별 $\hat{Q}(s, a)$ 값

	ω_i	MC1	MC2	MC3	MC4	MC5
a	0.6	3.0	6.0	3.6	3.6	2.4
b	0.3	3.0	1.8	1.8	1.2	1.5
c	0.1	0.5	0.0	0.6	0.7	0.9
\hat{Q}	-	6.5	7.8	6.0	5.5	4.8

선순위를 결정하기 때문에 작업물의 남은 공정시간을 1 순위, 다음 공정의 남은 작업시간을 2 순위 그리고 로봇의 이동시간을 3 순위로 결정하였다. 그리고 가중치의 합은 1로 하였고 모든 ω 값은 0 이상의 값을 갖도록 설정하였다.

식 (2)를 통하여 각 MC 별 \hat{Q} 값을 얻고 그중 제일 작은 값을 새로운 Q 값으로 식 (3)과 같이 계산하여 사용한다.

$$Q(s, a) = r + \text{Min} \hat{Q}(s', a') \quad (3)$$

Q-learning 알고리즘에서는 보상 값 r의 지속적인 갱신을 통하여 의사결정에 반영한다. 그러나 본 논문에서 사용하는 방법에서는 반송로봇이 한번 의사결정을 하여 시행하고 난 후 다음 의사결정을 내릴 순서가 되었을 때 기존의 값을 기억하고 있지 않아도 되기 때문에 보상 값 r은 0으로 설정한다.

표 4와 같이 MC별로 $\hat{Q}(s, a)$ 값을 계산한 결과 $\hat{Q}(2, 5)$ 값이 4.8로 $\hat{Q}(s, a)$ 값들 중 에서 제일 작은 것을 확인할 수 있었다. 따라서 반송로봇은 MC 5로 이동하여 작업 완료 신호가 발생할 때 까지 대기한다.

4. 사례연구(Case Study)

4.1 모의실험 모델(Simulation Model)의 구성

본 논문에서 구성한 모델의 Layout은 그림 2와 같이 두 개의 Zone으로 구분하여 각 Zone의 중심에 한 대의 로봇을 배치시킨 FMC를 구성하였다. 로봇의 반경을 고려하여 하나의 Zone에는 4대의 MC을 다른 하나의 Zone에는 3대의 MC을 배치하였고 Zone과 Zone을 연결시켜

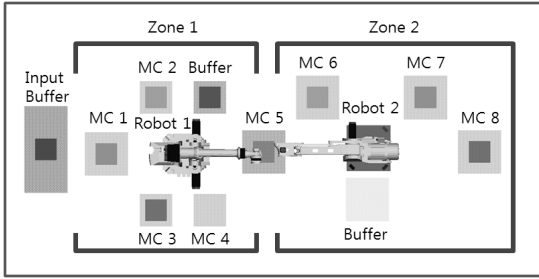


그림 2. FMC 시뮬레이션 모형

주는 가운데 부분에 한 대의 MC을 배치하였다. 작업물이 FMC로 투입되어 MC 1의 작업 유무에 따라 대기할 수 있는 Input buffer를 FMC의 맨 좌측에 배치하였고 각 Zone마다 MC의 작업 유무에 따라 대기할 수 있는 위치인 buffer를 두었다. 각 Zone에 배치된 로봇의 사항은 상의하며 각 MC들은 특정 작업시간을 가지고 작업물들이 해당 공정에 도착 하였을 때 공정이 시작되도록 하였다.

4.2 모의실험 모델의 가정

본 논문에서는 제한한 알고리즘의 유효성을 검토하기 위하여 시뮬레이션 분석을 한다. 따라서 시뮬레이션 분석을 위한 모델을 구성함에 있어 전제 조건이 필요하다. 다음은 시뮬레이션 모델의 전제 조건이다.

- Machine의 용량 : 1 capacity
- 제품의 종류 : 3 type
- 모든 제품의 도착 시간 : Exponential 40 minutes
- Robot 상차(Loading), 하차(Unloading)시간 : 10초
- Machine & Robot의 MTTR/MTBF 고려하지 않음
- Input Buffer & 각 Zone의 Buffer 용량 : Infinite
- Robot Hand의 용량 : 1 EA

다음은 위의 전제조건을 기반으로 작업물의 공정 순서와 각 MC에서의 작업시간, Robot의 이동시간을 설명한 내용이다.

모든 Machine의 작업 용량은 1로 설정하였고 각 Zone마다 하나의 버퍼를 두어 다음 공정의 MC가 가동 중 이라면 버퍼를 통해 대기한 후 다음 공정으로 연결되는 구조로 가정하였다. 작업물은 총 3 종류로 구성하였고 FMC에 지수분포 40분에 따라 input buffer를 통하여 시스템에 들어오게 되고 표 5와 같이 서로 다른 공정 순서를 거쳐 작업이 이루어지도록 구성하였다.

로봇을 통해 운반된 작업물들은 각각의 MC에서 특정

표 5. 작업물별 공정 순서

작업물 \ 순서	Part A	Part B	Part C
1	MC1	MC1	MC1
2	MC2	MC3	MC2
3	MC3	MC4	MC4
4	MC4	MC2	MC3
5	MC5	MC5	MC5
6	MC6	MC6	MC6
7	MC7	MC7	MC7
8	MC8	MC8	MC8

표 6. Machine별 작업시간

Machine 번호	작업시간 (단위 : 분)
MC1	N (10,2)
MC2	U (10,5)
MC3	N (5,2)
MC4	N (7,2)
MC5	U (10,2)
MC6	U (8,3)
MC7	N (10,2)
MC8	N (13,3)

작업시간을 가지게 되는데 해당 MC의 작업시간은 작업물의 종류와 관계없이 표 6과 같이 분포로 설정하여 작업의 유동성을 고려하였다. 모든 작업물은 MC와 MC사이를 이동할 때 해당 Zone의 중앙에 위치한 로봇에 의해 이루어진다. 로봇의 이동시간은 구성 된 MC들의 배치에 따라 표 7과 같이 구성하였으며 한 번에 하나의 작업물만 운반 할 수 있도록 하였고 작업물의 상차, 하차시간은 모두 10초로 가정하였다. 모든 공정을 거쳐 작업이 완료된 작업물은 MC 8을 마지막으로 시스템을 떠나게 된다.

4.3 로봇의 의사결정 알고리즘

로봇은 현재 진행 중인 작업을 완료하고 대기 상태가 되었을 때 Machine에서 작업완료 신호가 있는지 없는지 파악해야 한다. 신호가 있는 경우 중 작업 하나의 작업 완료 신호가 발생하였을 때에는 해당 작업을 수행하고 두개 이상의 신호가 발생하였을 경우에는 어떤 작업을 먼저 수행해야 할지 결정해야 한다. 또한 신호가 없는 경우에는 어느 작업물의 운반을 우선적으로 수행할지 결정해야 한

표 7. 로봇의 이동시간

Zone 1 (단위 : 초)						
	MC1	MC2	MC3	MC4	MC5	BF
MC1	-	30.0	30.0	45.0	60.0	45.0
MC2	30.0	-	35.0	40.0	45.0	30.0
MC3	30.0	35.0	-	30.0	45.0	40.0
MC4	45.0	40.0	30.0	-	30.0	35.0
MC5	60.0	45.0	45.0	30.0	-	30.0
BF	45.0	30.0	40.0	35.0	30.0	-
Zone 2 (단위 : 초)						
	MC5	MC6	MC7	MC8	BF	
MC5	-	20.0	60.0	70.0	45.0	
MC6	20.0	-	45.0	50.0	40.0	
MC7	60.0	45.0	-	25.0	40.0	
MC8	70.0	50.0	25.0	-	40.0	
BF	45.0	40.0	40.0	40.0	-	

표 8. 작업물의 시간 계산을 위한 Logic

```

Int Variable x (Robot availability)
Int Variable y(i) (Machine process time)
                (i is Machine number)
While do
    x ← y(i) - Simulation runtime
    Update x
    Wait
    if x <= 0
        break
repeat
    
```

다. 예를 들어 Zone1의 로봇은 모두 버퍼를 포함하여 5개의 작업장을 관할하고 있다. 선 작업을 완료하게 되면 로봇은 작업완료신호 발생여부를 확인한다. 작업 완료 신호가 없는 경우 5개의 작업장의 정보들을 이용하여 의사결정을 하고 작업 완료 신호가 두 개 이상 있는 경우 해당 작업장의 정보들만 이용하여 의사결정을 내린다. 의사결정을 하기 위해 필요한 정보들은 2장에서 언급한 현재 진행 중인 작업물들의 남은 작업 시간 정보와 작업물의 다음 공정의 남은 공정시간 정보와 로봇의 이동시간을 사용하였다. 또한 가중치는 3장에서 언급한 것과 같이 모든 MC의 작업시간을 1순위로 하여 0.6을 주었고 다음 MC의 남은 작업시간을 2순위로 0.3값을 주었고 마지막으로 로봇의 현재 위치에서 다른 MC으로의 이동시간을 3순위로 0.1을 주었다.

로봇은 위에서 얻은 정보들을 Q-learning 알고리즘을 이용하여 Q 값을 계산하고 의사결정을 내린다.

4.4 작업물의 신호 전달 알고리즘

작업물은 자신이 위치한 Machine에서 계속적으로 로봇에게 정보를 준다. 로봇에 의해 해당 Machine으로 운반되어 상차되면 진행해야할 공정시간을 먼저 확인한다. 그리고 공정이 시작되기 전 자신의 위치정보와 시뮬레이션상의 현재시간을 확인하여 지정된 변수에 입력하고 수행해야할 공정시간을 또 다른 변수에 입력하여 그 차이를 계산하여 지속적으로 정보를 제공한다. 그러므로 로봇은 자신이 의사결정을 해야 할 상태가 되었을 때 즉각적으로 정보를 사용할 수 있다.

5. 실험 및 결과

본 실험의 주목적은 한 대의 로봇을 중심으로 운영되는 영역의 FMC에서 로봇의 이동규칙과 로봇의 대기정책에 따른 수행도를 비교 평가하여 제시한 방법의 유효성을 검증하기 위한 것이다. 본 논문에서 제시한 방법의 유효성을 검증하기 위해서 하나의 다른 모델과 비교 하였다. 비교 대상은 로봇이 작업 완료 후 다음 완료 신호가 발생하지 않았을 때 선 작업을 마친 곳에서 대기하고 완료 신호가 하나 이상 발생하였을 때 먼저 발생한 작업물을 우선적으로 운반하는 로직을 반영한 모델(SPT)이다.

시뮬레이션 분석은 총 24시간 동안 시뮬레이션 모형을 가동시켰을 때 중요변수들의 수치로 판단하였다. 시뮬레이션 분석에 있어 모델은 처음 실행되었을 때 재공품이 존재하는 것이 아니라 시스템이 초기화된 상태에서 시작된다. 따라서 모든 로봇과 MC가 안정화된 상태가 될 때까지 준비시간이 필요하다. 따라서 본 논문에서는 FMC 내에 존재하는 모든 MC의 가동률을 기반으로 Warm-up 분석을 실시하였다. 실험 결과 약 1시간의 준비시간을 통해 시스템이 안정화되는 것을 확인할 수 있었기 때문에 준비시간을 1시간으로 설정하였다. 또한 분석의 유의성을 고려하여 5번의 반복 실험을 실시하였다. 시뮬레이션 분석에 사용되는 중요 변수로는 작업물이 FMC내에 Input Buffer에서 대기한 시간을 제외하고 작업물의 총 처리시간, 24시간동안 처리한 작업물의 수, Input Buffer에 대기하고 있는 작업물의 수, 로봇의 가동률을 사용하였다. 로봇의 가동률은 로봇이 가동되고 있는 시간을 비율로 나타낸 것이다. 표 9는 SPT규칙을 사용한 모델과 본 논문에서 제시한 방법을 사용한 모델과 비교 분석한 것이고 로봇의

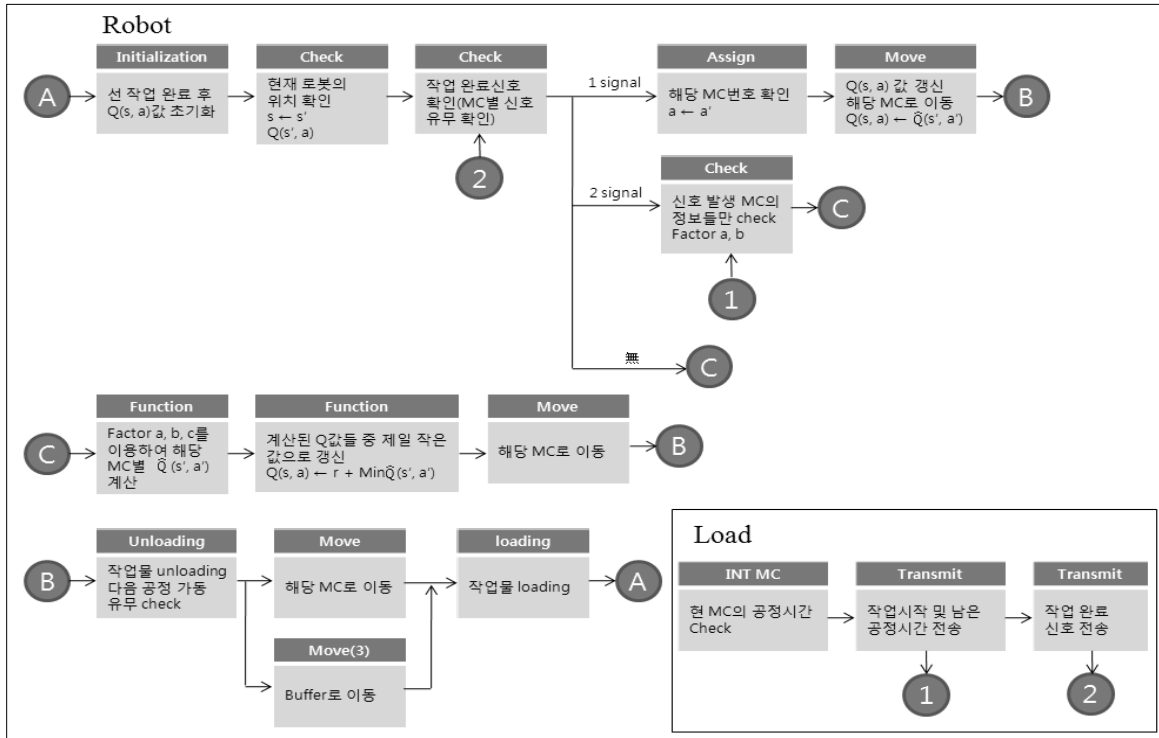


그림 3. 시뮬레이션 모델의 흐름도

표 9. 대안비교 결과

	SPT		Q-LN		단위
작업물의 총 처리시간	172.07		125.39		분
InputBuffer의 평균 대기 작업물의 수	10.34		6.62		EA
작업물의 총 처리량	89		101		EA
로봇의 가동률	1	2	1	2	%
	28.3	19.8	30.1	21.8	
	24.05		25.95		

가동률은 로봇 두 대의 값의 평균을 표시하였다. 비교 결과 Q-learning 알고리즘을 사용한 모델의 경우 작업물의 평균 체류시간이 125.39분이었다. 이 시간에 작업물이 FMC에 투입되어 모든 공정을 마칠 때 까지 MC별 최대 가공시간(약 84분)과 로봇에 의해 운반되는 시간(3.58분)을 제외하면 작업물이 로봇에 의해 운반되기 전까지 대기한 시간이 산출된다. 계산 결과 31.81분을 대기 한 것으로 나타났고 SPT법칙을 토대로 한 모델의 경우에는 84.49분

으로 약 50분의 차이가 나타남을 확인하였다.

또한 Input Buffer에 작업물의 평균대기수도 1.5배 이상 차이를 보였고 총 처리량도 12개 차이가 나는 것으로 확인되었다. 로봇의 가동률은 Q-learning 알고리즘을 사용한 모델의 경우 평균 25.95%, SPT법칙을 고려한 모델의 경우 24.05%로 약 2% 차이가 발생한 것을 확인할 수 있었다. 이는 선 작업을 수행한 후 작업 완료 신호가 발생하지 않은 경우 다음 작업을 선택하여 미리 그 위치로 이동하여 대기함으로써 높은 결과가 나타난 것으로 사료된다.

6. 결론

본 논문에서는 한 대의 로봇에 의해 제어되는 로봇 중심 FMC에서 로봇이 대기 상태가 되었을 때 작업물 선택 의사결정 방법으로 Q-learning 알고리즘을 응용한 방법을 제시하였다. 현재까지 연구된 로봇의 의사결정 방법은 간단한 시스템을 모델링하여 몇 개의 요소만을 고려한 수학적 기법이나 간단한 모델링을 통하여 이루어져 왔다. 하지만 본 논문에서는 작업 완료 신호가 2개 이상 있는 경우 해당 MC의 정보들만을 활용하여 작업을 선택하고 작업 완료 신호가 없을 시 자신의 영역에 있는 모든 MC

의 정보들을 활용하여 다음 운반해야 할 작업물을 미리 선택하여 해당위치로 이동 후 대기함으로써 전체 생산량을 증가시키는 방법을 제시하였다. 또한 의사결정을 하기 위한 요소를 본 논문에서는 3가지만 사용하였으나 여러 가지 요소를 사용하더라도 사용이 가능할 것이다.

본 논문에서 제시한 알고리즘은 현재 산업 분야 중에서 LCD생산 공정과 같이 거의 모든 공정이 반송용 로봇에 의해 구성되어 있는 장치 산업분야에서 사용한다면 큰 효과를 얻을 수 있을 것으로 사료된다.

향후 연구과제로는 로봇의 의사결정에 있어 단지 하나의 Zone에서 발생한 정보들을 활용하는 것이 아니라 다음 Zone의 로봇의 상태와 Machine의 정보와 로봇간의 interaction을 고려한다면 더 좋은 결과를 얻을 수 있을 것으로 사료된다.

참 고 문 헌

- 김주용, 강철구 (2005), “글래스 반송용 로봇핸드의 채터링 원인해석”, *한국정밀공학학회지*, 제22권, 제2호, pp. 98-104.
- 김재현, 서일홍 (2007), “지능형 로봇 시스템을 위한 영역 기반 Q-learning”, *대한전자공학회*, 대한전자공학회 학술 발표회 논문집, pp. 271-276.
- 박양병 (1992), “로봇 중심 FMC에서 최적 로봇 서비스 이동정책을 위한 연구”, *한국시물레이션학회지*, 제1권, 제1호, pp. 55-63.
- 서상욱, 심귀보, 김호덕 (2007), “군집 로봇의 협조 행동을 위한 강화 학습 기반의 진화 및 학습 알고리즘”, *한국퍼지 및 지능시스템학회 논문지*, 제17권, 제5호, pp. 591-597.
- 윤한열, 심귀보 (2005), “다수 로봇의 목표물 탐색을 위한 Area-Based Q-learning 알고리즘”, *한국퍼지 및 지능시스템학회 논문지*, 제15권, 제4호, pp. 406-411.
- 허병완, 이홍철 (2005), “Compromise Programming을 이용한 물류센터 설계에 관한 연구”, *한국시물레이션학회지*, 14권, 3호, pp. 43-54.
- Berenji, H. and Khedkar (1992), P., “Learning and tuning fuzzy logic controllers through reinforcement”, *IEEE Trans. on Neural Networks*, Vol. 3, No. 5, pp. 724-740.
- Brooks Automations, Inc. Factory Automation Software Division (2003), “AutoMod User’s Guide”, Brooks Automation, Inc.
- Carreras, M., Yuh, J. and Batlle, J. (2007), “Application of SONQL for real-time learning of robot behaviors”, *Robotics and autonomous systems*, Vol. 55 No. 8, pp. 628-642.
- Cogun, C. and Savsar, M. (2006), “Performance evaluation of a flexible manufacturing cell (FMC) by computer simulation”, *Modeling Meas Control B*, Vol. 62, No. 2, pp. 31-44.
- Er, M. J. and Zhou, Y. (2005), “Intelligent Fuzzy Q-Learning Control of Humanoid Robots”, *Lecture Notes in Computer Science*, Vol. 3498, pp. 216-221.
- Henneke, M. J. and Choi, R. H. (1998), “Evaluation of FMS parameters on overall system performance”, *Computers and Industrial Engineering*, Vol. 15, No. 1, pp. 324-330.
- Kiguchi Kazuo, Nanayakkara Thrishantha, Watanabe Keigo (2003), “Multi-Dimensional Reinforcement Learning Using a Vector Q-Net Application to Mobile Robots”, *International Journal of Control, Automation and Systems*, Vol. 1, No. 1, pp. 142-148.
- Oh, S. K, Kim, D. W. and Park, B. J. (2001), “Advanced Polynomial Neural Networks Architecture with New Adaptive Nodes”, *Transactions on control, automation and systems engineering*, Vol. 3, No. 1, pp. 43-50.
- Sabuncuoglu, I. and Homertzheim, D. L. (1989), “Expert simulation systems recent developments and applications in flexible manufacturing systems”, *Computers and Industrial Engineering*, Vol. 16, No. 4, pp. 575-585.
- Salichs, M. A., Puente, E. A., Gachet, D. and Pementel, J. R. (1993), “Learning behavioral control by reinforcement for an autonomous mobile robot,” *IEEE Conference on R&A*, Vol. 1, pp. 1436-1441.
- Savsar, M. and Cogun, C. (1993), “Stochastic modeling and comparisons of two flexible manufacturing cells with single and double gripper robots”, *International Journal of Production Research*, Vol. 31, No. 3, pp. 633-645.
- Savsar, M. and Aldaihani, M. (2008), “Modeling of machine failures in a flexible manufacturing cell with two machines served by a robot”, *Reliability engineering & system safety*, Vol. 93, No. 10, pp. 1551-1562.



윤 정 익 (rayheart@korea.ac.kr)

2005 한남대학교 산업시스템공학과 학사
 현재 고려대학교 정보경영공학부 석사과정

관심분야 : Discrete Event Simulation, Digital Manufacturing, SCM



엄 인 섭 (uis27@korea.ac.kr)

2002 고려대학교 산업시스템정보공학과 학사
 2005 고려대학교 산업시스템정보공학과 석사
 현재 고려대학교 정보경영공학부 박사과정

관심분야 : 생산 물류 정보 시스템, Discrete Event Simulation, SCM, PLM



이 흥 철 (hlee@korea.ac.kr)

1983 고려대학교 산업공학과 학사
 1988 Univ. of Texas 산업공학과 석사
 1993 Texas A&M Univ. 산업공학과 박사
 현재 고려대학교 정보경영공학부 교수

관심분야 : SCM, 생산 및 물류 정보시스템, PLM