

다 개체 이동 로봇의 협동 제어

Cooperative Control of Multiple Mobile Robots

○이경노*, 이두용**

* 한국과학기술원 기계공학과(Tel : +82-42-869-3269; Fax : +82-42-869-3210; E-mail: jeje@discrete.kaist.ac.kr)

** 한국과학기술원 기계공학과(Tel : +82-42-869-3229; Fax : +82-42-869-3210; E-mail: leedy@cais.kaist.ac.kr)

Abstract This paper presents a cooperative control method for multiple robots. This method is based on local sensors. The proposed method integrates all information obtained by local perception through a set of sensors and generates commands without logical conflicts in designing control logic. To control multiple robots effectively, a global control strategy is proposed. These methods are constructed by using AND/OR logic and transition firing sequences in Petri nets. To evaluate these methods, the object-searching task is introduced. This task is to search an object like a box by two robots and consists of two sub-tasks, i.e., a wall tracking task and a robot tracking task. Simulation results for the object-searching task and the wall tracking task are presented to show the effectiveness of the method.

Keywords Cooperative Control, Multiple Robots, Mobile Robot, Petri nets, Simulation

1. 서론

협동 로봇 시스템은 여러 대의 로봇들이 역할 분담과 기능 통합을 통해 목표를 완수하는 로봇 시스템으로, 강건성, 유연성, 저 단가, 소형, 협동을 통한 성능 향상 등의 장점이 있다.[2][3][9] 따라서, 협동 로봇 시스템은 물체 이송[2][4][6][9], 청소 작업[1][7], 생산, 의료, 서비스 분야 및 극한 환경 등 그 응용 잠재력이 매우 크다. 그러나, 협동 로봇 시스템은 실제적으로, 로봇 수의 증가에 따른 상황 변화와 로봇 간의 상호 작용의 영향이 기하급수적으로 증가하여 해석이 어려운 시스템이다.[8] 또한 여러 대의 자율 이동 로봇이 협동 작업을 하기 위해서는 상대 로봇의 위치와 움직임을 파악, 그에 따라서 각자의 주행을 정확히 제어하는 기술이 필요한데, 이동하는 타 로봇의 위치 정보와 이동 변화의 추이를 정확히 획득한다는 것은 매우 어려운 문제인 것이다. 특히, 여러 로봇들이 주위 환경을 인지하고 상호 협동 작업을 행하는 과정에서 직접적인 통신을 이용하지 않고, 지역적인 센싱을 이용하여 협동 제어하는 경우 라던 센싱 정보의 불확실성과 애매성 등에 의해서 여러 로봇들이 협동할 수 있는 제어 로직의 설계는 더욱 어려워지게 된다.

이러한 문제를 해결하고자, 본 연구에서는 지역적 센싱에 의존하는 로봇들이 타 로봇의 위치 정보와 이동 변화의 추이를 페트리넷에 의한 제어를 통해 인지하고, 따라서 효과적인 협동 작업을 수행하도록 하고자 한다. 그리고, 로봇의 작업에 근거한 총괄적 제어 알고리즘의 설계 방법을 제시하고, 이를 이용하여 각 로봇에게 적절한 작업을 할당한다. 본 연구에서 개발된 방법론의 검증은 위해서 두 대의 로봇이 제한된 영역에서 역할 분담과 타 로봇의 행동 추이를 통해 추적 주행하면서 원하는 대상을 찾는 작업을 설정하고 시뮬레이션을 통해 검증하였다. 그런데, 이러한 협동 작업 수행을 위한 제어 로직을 구성함에 있어서, 로봇에 장착된 많은 센서들로부터의 지역적인 센싱 정보들은 매우 다양하고, 그 수도 많기 때문에 센싱 정보 중 유사 상황을 하나로 통합하여 제어를 하는 것이 더 편리하다. 따라서 이러한 지역적 센싱 정보 중 유사 상황을 하나로 통합하는 방법과 제어 명령들간의 충돌 방지를 위해 페트리넷을 이용한 방법론을 제시한다. 센싱 정보의 통합 방법과 제어 명령 생성 방법의 타당성을 검증하고자 8개 방향에 8개 센서를 가진 단일 로봇이 지역적 센싱을 통해 벽을 감지하고, 충돌이나 벽으로부터의 이탈 없이 벽을 추적하는 작업을 설정, 시뮬레이션을 통해 그 타당성을 입증하였다.

2장에서는 센싱 정보 통합 방법과 제어 명령 생성 방법에 대해서 살펴보고, 3장에서는 로봇의 작업 상황에 따른 두 대 로봇의 총괄적 제어 로직 설계 방법을 제시한다. 4장에서는 2장, 3장에서 설명한 방법론의 검증은 위한 시뮬레이션의 방법을 설명하고 5장에서는 시뮬레이션 결과를 제시한다.

2. 센싱 정보 통합 방법 및 제어 명령 생성 방법

각 로봇들에 장착되어 있는 센서들을 통해서만, 작업 환경에 대한 정보 습득 및 상황 인지를 하게 되는 경우에는 충분한 정보를 얻어내고 정교한 작업을 수행하기 위해서 센서의 수가 많아지게 되며, 센서 시스템도 복잡해진다. 따라서 센싱 정보 해석의 복잡도는 그만큼 더 증가하게 되는데, 이런 시스템의 제어 로직을 구성할 때, 'if-then'을 이용하여 제어 로직을 구성하면 유사하게 해석될 수 있는 센싱 정보들을 하나로 통합하여 명령을 내리도록 구성하는 것이나, 복합적인 로봇 상황으로 인하여 발생할 수 있는 제어 로직간의 충돌 방지를 위한 로직간의 선후 관계 형성을 체계적으로 접근하기가 어렵다. 따라서, 본문에서는 이러한 문제를 해결하기 위해서 페트리넷을 이용한 제어 로직 구성의 방법을 제안하고자 한다.

우선, 유사한 센싱 정보를 하나로 통합하는 경우에 대한 페트리넷 모델은 다음과 같은 방법으로 설계된다. 센싱 정보를 페트리넷에서의 하나의 place로 모델링한다. 그런 후에, 로봇의 상황에 따라 발생할 수 있는 센싱 정보의 상황을 고려해서 유사한 상황은 페트리넷의 AND/OR 로직을 이용해서 또 다른 하나의 place로 통합한다. 통합된 place에 대해서는 출력 transition에 로봇 상황에 맞는 제어 로직을 부과하고, 이를 점화시킴으로써 로봇을 제어하게 되고, 로봇은 다른 상황으로 전환되게 된다. 이렇게 센싱 정보를 통합하고, 그에 맞는 제어 로직을 결정하는 과정에서 일부 제어 로직간의 충돌이 생겨나게 된다. 이러한 충돌 문제를 해결하기 위해 사건의 중요성에 따라 차례대로 transition 번호를 붙여 transition 집합의 우선 순서를 결정하고, 충돌 시 transition 번호가 빠른 것을 우선적으로 점화하여 충돌 문제를 해결하게 된다. 이렇게 점화 가능한 transition 번호의 순서에 따라서 점화를 하면, 결국 하나의 제어 명령만이 생성된다. 본 모델에서의 모든 transition은 immediate transition이다. 이러한 제어 로직 설계의 방법은 하나의 특정 작업에 대한 제어 로직을 설계하는 방법이며, 본 방법은 단일 로봇이 8개 방향의 8개 센서에 근거하여 벽을 추적하는 작업에 대한 시뮬레이션을 통해 검증된다.

페트리넷을 이용한 로직 설계의 방법은 페트리넷이 그래픽적인 도구이므로, 유사 상황들을 하나의 상황으로 통합하는 과정이 일목요연하게 보여짐으로써, 유사 상황들 중에서 중복이나 누락의 상황 발생을 없앨 수 있으며, 마찬가지로 중복이나, 상충된 제어 명령을 사전에 검증, 제거할 수 있다. 또한, 페트리넷 모델링 시뮬레이션을 통해 센싱 정보들의 변화에 따른 제어 로직 변화에 대한 추적이 용이하고, 복잡한 센싱 정보에 의해서 발생하는 제어 명령간의 충돌을 transition 번호의 우선 순위로 간단히 해결할 수 있다는 장점이 있다. 앞서 설명한 방법론을 검증하기 위한 단일 로봇의 벽 추적 작업에 대한 페트리

넷 모델링 및 시뮬레이션 결과를 5장에서 제시한다.

3. 로봇 군의 총괄적 제어 방법

본 연구에서 제시하는 총괄적 제어는 단순히 로봇에게 작업을 할당하는 역할만을 담당한다. 이 방법은 2장에서의 방법론에 의해서 개발된 로봇의 특정 작업의 수행에 대한 독립적인 제어 로직을 각 로봇에게 지정해주는 것으로, 각 로봇들이 독립적인 작업 수행을 통해 공동의 목표를 달성하도록 해주는 역할을 담당하는 것이다. 이 방법은 작업을 각 로봇에게 단순히 지정하는 방식이므로, 통신의 방법을 사용하기는 하나, 로봇 수의 증가에 대해서 통신 프로토콜이 복잡해진다거나, 통신의 병목현상이 생긴다거나 하지 않는다. 본 제어 법은 로봇들의 현재 작업을 쉽게 파악할 수 있어야 하고, 각 로봇의 특정 작업 조건이 만족될 때에는 작업 전환의 명령을 지시하도록 알고리즘 설계를 해야 하므로, 페트리넷을 이용하는 방법을 제시한다.

우선, 공동의 목표 작업을 한 대의 로봇이 행할 수 있는 세부 작업들로 세분화한다. 그리고, 각각의 세부 작업들을 하나의 로봇 상황에 간주하여, 페트리넷의 place로 모델링한다. 세부 작업들간의 상호 관계 및 흐름 절차에 따라 페트리넷 모델링을 완성한다. 그리고, 페트리넷 모델링의 토큰(token) 하나는 로봇 한대로 모델링한다. 이 토큰에 실제 각 로봇에게 부과시킨 로봇 번호와 같은 번호를 부착시킴으로써, 페트리넷 모델링에서의 토큰의 흐름 파악을 통해 각 로봇의 상황을 쉽게 파악할 수 있다. 또한, 특정 번호가 붙은 토큰의 위치를 transition의 점화에 의해서 여러 가지 다른 세부 작업의 place들로 이동시킴으로써, 로봇의 작업을 전환시킬 수 있게 된다. 이러한 제어 전략을 설계하는 과정에서 생겨나는 제어 명령간의 충돌은 앞서 설명한 사건 우선 순서에 의한 transition 번호 부여 법에 의해서 해결한다.

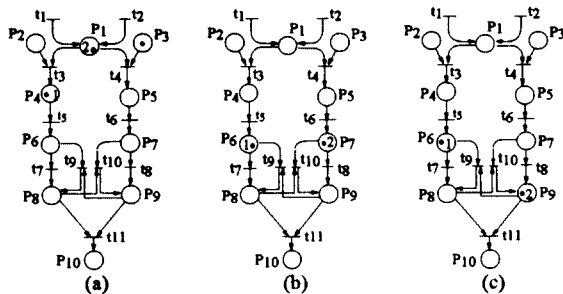


그림 1. 로봇 두 대의 총괄적 제어에 대한 페트리넷 모델링
Fig 1. Petri nets modeling for global control strategy with two robots

표 1. 총괄적 제어에 대한 페트리넷 모델링 리스트
Table 1. Petri nets modeling for global control strategy

Places		Transitions	
P1	초기 상태	t1, t2	시작
P4, P5	하부 작업(A,B) 인지	t3, t4	로봇에 하부 작업 지정
P6, P7	하부 작업의 수행 중	t5, t6	하부 작업 수행 시작
P8, P9	하부 작업의 완수	t7, t8	하부 작업 수행 종료
P10	공동의 작업 완수	t9, t10	수행 중인 작업 종료

그림 1은 앞서 설명한 방법론에 의해서, 두 대의 로봇이 공동 목표를 달성하기 위해 세분화시킨 하부 작업(A,B)을 수행하고, 두 대 중 어느 한 대가 목표를 달성하면, 다른 한 대가 수행 중인 현재의 하부 작업을 강제 종료시키는 작업에 대한 페트리넷 모델링과 그 작업 진행의 흐름 제어를 보여주고 있다. 그림 1(a)는 초기 상황에서 각 로봇에게 하부 작업을 할당해주는 과정을, 그림 1(b)는 두 로봇이 각각 다른 하부 작업을 수행 중임을, 그림 1(c)는 2번 로봇만이 목표를 달성, 1번 로봇의 작업을 강제 종료시키는 제어 흐름을 보여준다. 이러한 작업 전환에 대한 총괄적 제어의 방법은 5장에서 두 대의 로봇이 제한된 영역 내의 대상물을 효과적으로 찾기 위해 벽 추적 작업과 로봇 추적 작업으로 각각 나눠 실행하는 작업에 대해서 구

체적으로 적용되고, 시뮬레이션을 통해 평가된다.

앞서 살펴본 바와 같이, 페트리넷으로 모델링하는 경우에는 가시적인 효과에 의해 각 로봇의 현재 작업 상황을 쉽게 파악할 수 있으며, 따라서 다른 작업으로의 전환 계획이나 제어 명령 구조의 형성이 용이하다. 특히, 앞 장에서 설명한 페트리넷에 기반한 제어 알고리즘의 설계와 직접적인 연계가 가능하므로, 하부 작업의 제어 알고리즘을 페트리넷으로 설계할 경우, 총괄적인 제어에서부터 하부 작업에서의 세부 동작에 이르는 제어 계획의 일원화와 체계화를 이루어낼 수 있는 장점을 가지고 있다.

4. 시뮬레이션 환경

본 시뮬레이터는 2장과 3장에서 설명한 방법론의 검증을 위해서 선정한 단일 로봇의 8개 방향, 8개 센서에 근거한 벽 추적 작업, 다른 로봇의 추적 주행 작업, 두 대 로봇의 벽 추적 작업과 로봇 추적 작업으로의 역할 분담을 통한 대상물 찾기 작업의 시뮬레이션을 위해서 개발되었으며 물리적 환경과 비슷하도록 제한 조건을 다음과 같이 전제한다.

4.1 시뮬레이션 환경 내의 로봇 시스템

시뮬레이션 내의 작업 환경 및 로봇은 다음과 같다.

- (1) 작업 환경의 바닥은 마찰이 일정하며 동일하다.
 - (2) 시뮬레이션 프로그램은 C-language로 짜여졌으며, PC에서 구동한다.
 - (3) 자율 이동 로봇은 단지 두 개의 DC-motor와 연계된 좌·우 바퀴와 캐스터 한 개로 구성된다. 따라서, 로봇 시스템은 구조적으로 nonholonomic 특성을 갖는다.
 - (4) 로봇이 행할 수 있는 동작은 좌·우 두 바퀴의 속도에 의해서 '전진, 후진, 좌·우 방향으로 회전하기(left/right turning)', 좌·우 방향으로 제자리 회전하기(left/right rotation), 정지, 대기 상태(idle)'의 8가지로 구분된다. 여기서, turning과 rotation의 구분은 회전 반경의 유무이다.
 - (5) 로봇의 판성 중심과 무게 중심은 로봇 중앙에 있다.
 - (6) 로봇은 초기 각도와 초기 위치를 알지 못한다.
 - (7) 로봇의 전체 크기는 23cm*23cm 정도의 크기이며, 바퀴의 크기는 직경 15cm, 양 바퀴간의 폭은 20cm, 최고 속도는 35cm/s, 최대 가속도는 25cm/s²으로 설정하였다.
 - (8) 로봇끼리 정보 교환을 할 수 있는 간단한 시스템이 있다.
- 로봇이 위와 같은 조건을 가지고 있으므로, 좌·우 바퀴의 속도에 의해서 동작이 결정되고 진행된다.

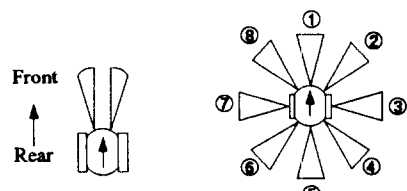
4.2 시뮬레이션 환경 내의 센서 시스템

기본적인 작업을 위해서 각 로봇은 타 로봇을 감지하는 센서, 경계 벽 감지 센서, 그리고 작업 환경 내에서의 대상물을 찾기 위한 센서들을 가지고 있다. 각 센서의 센싱 영역과 그 개수, 종류는 표 2에서 보여주는 바와 같다. 그리고 각 센서의 위치는 그림 2에서 보여주는 바와 같이 배치되어 있다.

표 2. 센서 시스템 리스트
Table 2. Sensor Systems List

센서 종류	센싱 대상	센싱 각도	센싱 거리	개수
초음파 센서	경계 벽	±10° 이내	75cm 이내	8개
	타 로봇	±15° 이내	75cm 이내	8개
적외선 센서	대상 물체	±10° 이내	75cm 이내	2개

여기서, 대상 물체는 box 형태의 물체를 의미한다.



(a) 대상물 감지 센서 (b) 로봇 및 벽 감지 센서

그림 2. 시뮬레이션 환경 내의 센서 시스템
Fig 2. Sensor Systems in Simulation Environment

각 센서의 최대 센싱 거리와 최대 센싱 각은 실제 센서가 허용하는 범위 내에서 선정된 것이다. 각 센서간의 간섭으로 인한 센싱 오류 및 센싱 여부의 판단이 어려운 상황이 발생하는 경우를 없애기 위해서 Kub6[6]가 도입한 직교 센싱(orthogonal sensing)의 개념을 이용한다.

5. 시뮬레이션 결과 및 분석

본 시뮬레이션은 두 대의 로봇이 8개 방향의 8개 센서로 주위 벽을 감지하여 원활하게 벽을 추적하는 벽 추적 작업과 또 다른 8개 방향 8개 센서를 이용, 이동 중인 타 로봇을 추적하는 작업을 이용하여, 장애물이 없는 제한된 영역 내에서 지역적 센싱을 이용한 대상물을 찾는 협동 작업의 원활한 수행을 입증해 보이려고 한다. 벽 추적 작업, 로봇 추적 주행 작업 및 작업 할당을 위한 총괄적 제어 방법을 차례로 설명하고, 벽 추적 작업 설명에는 2장에서 설명한 방법론의 검증에 위한 시뮬레이션 결과도 함께 제시한다. 그리고, 대상물을 찾는 작업에 대한 시뮬레이션 결과도 함께 제시된다.

5.1 벽 추적 작업에 대한 시뮬레이션

벽 추적 작업은 로봇에 장착된 센서의 수가 8개로 유사 상황으로 예측할 수 있는 센서의 상황이 다양하고, 센서의 센싱 영역이 제한적이기 때문에 기본적인 제어 알고리즘에서의 제어 명령 상충 상황이 발생하기 쉬워 센싱 정보 통합 법과 제어 명령 생성 법을 검증하기에 적합한 작업이다. 또한 효과적인 탐색 면적을 확보하므로 대상물을 찾는 작업의 하부 작업으로도 적합하다. 본 작업의 목적은 임의의 초기 위치, 각도를 가진 로봇이 벽과의 충돌 없이, 벽과 일정 거리를 유지하면서 반시계 방향으로 진행하는 것이다. 이를 위한 센서 시스템은 그림 2에서 보여주는 바와 같이 8개 방향에 각각 하나씩의 초음파 센서가 장착된 시스템이다.

다음은 벽 추적 작업의 기본적인 알고리즘이다.

- (1) 벽에 접근하려 할 때는 좌회전한다.
- (2) 벽에서 멀어지려 할 때는 우회전한다.
- (3) 벽에서 적정 거리를 유지할 때에는 직진한다.
- (4) 정면에 벽이 있는 경우는 큰 각으로 좌회전한다.
- (5) 좌측에 벽이 있는 경우는 정지 후, 우회전한다.
- (6) 벽을 감지 하지 못하면, 벽이 감지될 때까지 직진한다.

이러한 기본 알고리즘을 토대한 설계한 페트리넷 제어 로직은 다음과 같다.

표 3. 벽 추적에 대한 페트리넷 모델링 리스트
Table 3. Petri nets modeling list for wall-tracking control

Places			
P ₉	정지 상태	P ₁₄	대기 상태
P ₁₀	좌측에 벽이 존재 상황	P ₁₉	벽 감지에 실패
P ₁₁	정면에 벽이 존재 상황	P ₂₀	좌회전하기
P ₁₂	우측에 벽이 존재 상황	P ₂₁	직진하기
P ₁₃	뒤에 벽이 존재 상황	P ₂₂	우회전하기

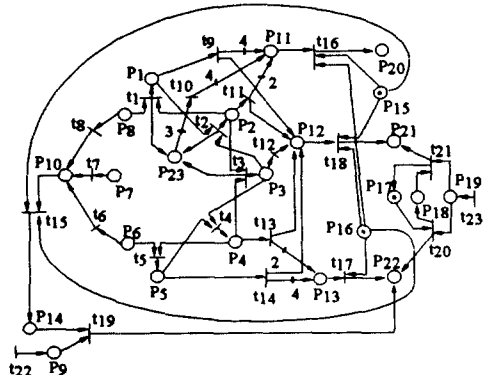


그림 3. 벽 추적 제어 알고리즘에 대한 페트리넷 모델링
Fig 3. Petri nets modeling for wall-tracking control

그림 3에서 Place P₁ ~ P₈은 각각 센서 1번~8번 센서의 상태이며, 벽을 감지하면 토큰(token)을 갖는다. 이것은 제어 로직 결정의 초기에 센서의 상황에 따라 주어진다. 그림 3의 transition t₁~t₁₄는 유사 센싱 상황을 통합하기 위해서 페트리넷의 AND/OR 로직을 이용한 것이다. 즉, 센서 8번과 센서 2번이 벽을 감지하면, 센서 1번이 벽을 감지 하지 못하더라도, 정면에 벽이 있는 상황으로 판단하여, 센서 1번이 벽을 감지한 상황과 동일시하고자 t₁을 이용해서 통합한다. 표 4는 이러한 센싱 정보 통합 계획의 일부를 보여주고 있다.

표 4. 센싱 정보의 통합 계획의 일부
Table 4. Portion in sensing information integration plan

유사 상황 Places	합쳐진 Place
P ₁ & P ₃	P ₂
P ₈ & P ₂	P ₁
P ₆ (OR) P ₇ (OR) P ₈	P ₁₀
P ₄ (OR) P ₅	P ₁₃

그림 3의 Transition t₁₅, t₁₆, t₁₇, t₁₈, t₂₀, t₂₁은 제어 명령을 생성하는 transition으로 각각 정지, 좌회전, 우회전, 직진, 우회전, 직진의 제어 명령을 의미한다. 이 transition의 번호 15, 16, 17, 18, 20, 21은 제어 알고리즘의 충돌 문제를 해결하기 위한 표 5의 사건 우선 순위에 근거하여 최우선 순위가 가장 빠른 번호를 갖도록 지정하여 transition 번호순에 의해서 제어 명령 생성 시 가장 우선적으로 실행되도록 한 것이다.

표 5. transition 점화 순서 결정을 위한 사건 우선 순서
Table 5. Event Priority for determining the transition firing sequence

Order	Event
1	로봇 좌측에 벽이 있을 경우
2	로봇 정면에 벽이 있을 경우
3	로봇이 벽에서 멀어지는 경우
4	로봇이 벽으로 접근하는 경우
5	벽 감지 실패한 경우

그리고 그림 3의 Transition t₂₀, t₂₁, t₂₃은 8개 센서 모두 벽을 감지 하지 못한 경우에 대한 제어 모델링이다.

그림 4는 임의의 초기 상황에 대한 시뮬레이션 결과 중 일부를 보여주고 있다. 그림 4는 초기에 정면과 좌측에 벽을 감지하고 있는 상태이므로, 두 상황에 대한 제어 알고리즘의 충돌이 예상된다. 시뮬레이션 결과는 이러한 충돌 상황이 해결되어 벽 추적이 계획대로 수행되고 있음을 보여준다.

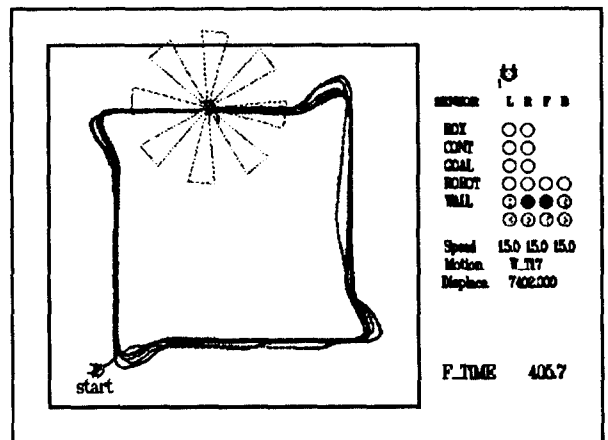


그림 4. 벽 추적 제어에 대한 시뮬레이션 결과
Fig 4. Simulation results for wall-tracking control

5.2 타 로봇 추적 작업

다른 로봇의 정확한 위치와 속도, 진행 방향을 모르는 상태에서 지역적 센서들로부터의 정보와 통신을 통한 간단한 정보만을 통해서 상대방 로봇의 위치를 추정, 타 로봇 및 벽과의 충돌 없이 벽 추적 로봇과 수평으로 일정 거리를 유지해야 한다.

본 작업 수행을 위한 제어 알고리즘은 로봇간의 거리, 로봇을 감지한 센서의 번호 등을 고려하여 로봇이 충돌하는가, 로봇이 서로 멀어지는가 등을 판단한다. 이러한 제어 알고리즘의 설계는 앞의 2장에서 설명한 센싱 정보의 통합 방법이나, 제어 명령의 생성법을 이용하며, 벽 추적 작업의 제어 로직 설계와 같은 방법으로 설계한다.

5.3 대상물 찾기 작업에 대한 총괄적 제어 계획

본 제어 계획은 앞의 3장의 방법에 근거한 것으로 두 대 로봇의 협동 작업을 이끌어 내기 위한 것이다. 본 작업의 목표는 장애물이 없는 한정된 공간 내에서 로봇이 초기 위치, 각도를 모르는 상태에서 출발하여 원하는 목표물을 찾는 것을 목표로 한다. 특히, 로봇 자신의 현재 위치를 모른다고 전제를 하였으므로, 두 로봇 모두 목표물을 찾기 위해서는 두 로봇이 함께 움직이는 효과적이다. 따라서, 하부 작업으로 벽 추적 작업에 의한 바깥쪽 탐색과 로봇 추적 주행 작업에 의한 안쪽 탐색의 두 가지로 구분하였다. 이 두 작업을 각 로봇에게 지정하고, 상황에 따라서는 두 로봇의 작업을 서로 바뀌거나, 종료시키는 작업으로 전환시키는 등 총괄적 제어 계획이 필요하게 된다. 그림 5는 이런 목적으로 계획된 제어 모델이다. 본 총괄적 제어의 알고리즘은 각 로봇에게 하부 작업을 부과하고, 진행 중 어느 한 로봇이 대상물을 찾으면, 다른 로봇은 로봇 추적 작업으로 전환하여 대상물을 찾은 로봇을 이용하여 자신도 대상물을 찾는다. 작업 진행 중에 두 로봇의 현 작업을 서로 교환하는 것이 더 효과적인 위치 배정 상태가 되면, 이를 판단하여 작업 교환을 하게 하는 제어 방법도 함께 계획되었다.

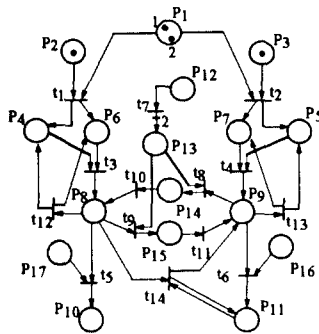


그림 5. 물체 찾기 작업에 대한 총괄적 제어 페트리넷 모델링
Fig 5. Petri nets modeling for global control in object-searching task

표 6. 총괄적 제어의 페트리넷 모델링 리스트
Table 6. Petri nets modeling list for global control strategy

Places			
P ₁	초기 상태	P ₈ , P ₉	선정 작업 진행 중
P ₄ , P ₅	물체 찾기 중	P ₁₀ , P ₁₁	작업 종료
P ₆	벽 추적 중	P ₁₂	로봇 간의 작업 교환 요구
P ₇	로봇 추적 중	P ₁₄ , P ₁₅	두 로봇의 작업 교환 중
P ₁₆ , P ₁₇	물체 찾은 상태		

5.4 대상물 찾기 작업에 대한 시뮬레이션 결과

벽 추적 제어, 로봇 추적 주행 제어에 대한 페트리넷 제어 알고리즘과 총괄적 제어 방법을 통한 각 로봇의 작업 할당을 통해, 두 대 로봇이 협동으로 대상물을 찾는 작업을 수행하였다. 로봇들의 초기 각도 및 대상물의 위치, 각도는 임의로 지정하였으며, 시뮬레이션 결과는 그림 6 과 같이 협동 작업을 계획대로 수행한다. 그림 6에서의 로봇의 이동 경로를 보면, 처음에는 1번 로봇이 2번 로봇을 추적하는 작업을, 2번 로봇이 벽을 추적하는 작업을 할당받았음을 알 수 있고, 그것이 그림 6

의 현 상태에서 서로 작업 교환이 이루어져 있음을 알 수 있다. 이것은 두 로봇의 총괄적 제어가 효과적으로 이루어지고 있다는 것을 보여준다.

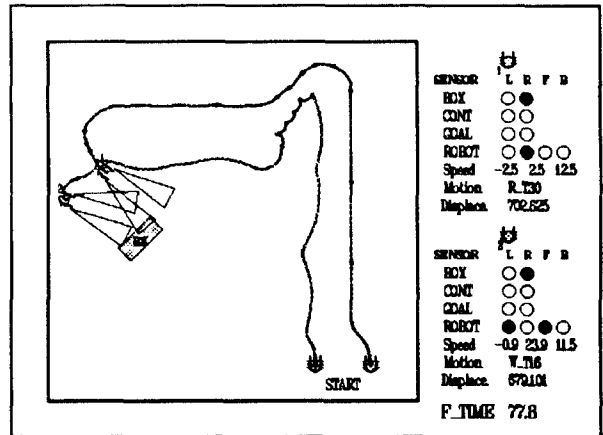


그림 6. 물체 찾는 협동 작업에 대한 시뮬레이션 결과
Fig 6. Simulation result for the cooperative task

6. 결론

본 연구에서는 지역적 센싱에 의존해서 협동 작업을 수행하는 다중 로봇 시스템 구현 시 문제가 되는 센싱 정보의 통합 방법 및 제어 명령 생성 방법에 대하여 페트리넷을 이용하는 방법을 제시하였으며, 그 효율성 및 가능성을 시뮬레이션을 통하여 제시하였다. 또한, 특정 작업에 대해서 설계된 제어 알고리즘을 협동 작업을 위해 각 로봇에게 부여해 주는 총괄적 제어 방법을 제시하였다. 그리고 제시된 방법들을 통하여 타 로봇의 위치와 행동 추이를 정확히 파악하기 힘든 지역적 센싱에 기반을 둔 두 대 로봇이 협동 작업을 원활히 수행할 수 있음을 제시하였다.

7. 참고 문헌

- [1] Tamio Arai, Daisuke Kurabayashi, Jun Ota, and Shinpei Ichikawa, "Motion Planning for Cooperative Sweeping with Relocating Obstacles", *Proc. of IEEE Int. Conf. on SMC*, 1996, p1513-p1518
- [2] John S. Bay, "Design of the "Army-Ant" Cooperative Lifting Robot", *IEEE Robotics and Automation Magazine*, March 1995, p36-p43
- [3] Y. Uny Cao, Alex S. Fukunaga, Andrew B. Kahng, Frank Meng, "Cooperative Mobile Robotics : Antecedents and Directions", *Proc. of IEEE Int. Conf. on IROS*, vol 1, 1995, p226-p234,
- [4] B. R. Donald, J. Jennings, D. Rus, "Analyzing Teams of Cooperating Mobile Robots", *Proc. of IEEE Int. Conf. on R&A*, pt 3, 1994, p1896-p1903
- [5] Toshio Fukuda, Go Iritani, "Construction Mechanism of Group Behavior with Cooperation", *Proc. of IEEE Int. Conf. on R&A*, Vol 3, 1995, p535-p542
- [6] C. R. Kube, H. Zhang, "The Use of Perceptual Cues in Multi-Robot Box-Pushing", *Proc. of IEEE Int. Conf. on R&A*, 1996, p2085-p2090
- [7] Daisuke Kurabayashi, Jun Ota, Tamio Arai and Eiichi Yoshida, "Cooperative Sweeping by Multiple Mobile Robots", *Proc. of IEEE Int. Conf. on R&A*, 1996, p1744-p1749
- [8] M. J. Mataric, "Issues and approaches in the design of collective autonomous agents", *Robotics and Autonomous Systems*, 16, 1995, p321-p331
- [9] Lynne E. Parker, "Designing Control Laws for Cooperative Agent Teams", *Proc. of IEEE Int. Conf. on R&A*, 1993, p582-p587
- [10] J. L. Peterson, *Petri Net Theory and the Modeling of Systems*. Englewood Cliffs, NJ : Prentice - Hall, 1981