

동적 환경에서 자율 이동 로봇군의 이동을 위한 신경 회로망 기반 인공 생명 모델

An Artificial Life Model Based on Neural Networks for Navigation of Multiple Autonomous Mobile Robots in the Dynamic Environment

민 석 기, 강 훈
(Suk-Ki Min, and Hoon Kang)

Abstract : The objective of this paper is, based upon the principles of artificial life, to induce emergent behaviors of multiple autonomous mobile robots which complex global intelligence form from simple local interactions. Here, we propose an architecture of neural network learning with reinforcement signals which perceives the neighborhood information and decides the direction and the velocity of movement as mobile robots navigate in a group. As the results of the simulations, the optimum weight is obtained in real time, which not only prevent the collisions between agents and obstacles in the dynamic environment, but also have the mobile robots move and keep in various patterns.

Keywords : artificial life, autonomous mobile robots, neural networks, flocking, obstacle avoidance

I. 서론

자율 분산 이동 로봇 시스템은 센서로부터 주위의 상황을 감지하여 그 상황에 맞는 행동을 취해 원하는 목표를 이루는 시스템이다. 지금까지의 중앙 관리형의 제어 구조를 가진 시스템에 비해 이 시스템은 자율적으로 행동을 정하는 개체가 다수 모여서 질서를 만들며 하나의 시스템을 구성하는 시스템 형태의 새로운 제어구조를 가지고 있다. 이러한 시스템은 자율 이동 로봇의 개수가 증가하더라도 전체 시스템의 복잡도는 증가하지 않는 장점을 가지고 있으며 다양한 작업을 수행하기 위해 서로의 기능을 보완하면서 협동하고 공동작업을 통해 주어진 작업을 수행하는데 효율적인 기능을 지니고 있다. 자율 이동 로봇은 스스로 행동을 결정하는 제어구조를 가지고 있음으로써 중앙의 제어 없이 로봇간의 협조 행동을 이끌어 낼 수 있고 분산된 시스템 특성을 가지고 있음으로써 로봇이 한 두대 고장 나더라도 전체 목표를 수행하는데 큰 지장이 없으며 기본적인 기능만으로도 구현할 수 있기 때문에 로봇의 제작비용이 적다는 장점을 가지고 있다. Brooks는 복수의 자율 이동 로봇이 단순한 행동 기반의 제어 구조를 가지고 작업을 수행하는 문제를 제기했다[1].

최근의 복수의 자율 이동 로봇 시스템에 대한 관심은 개체간 국부적 상호 작용에 집중되어있다. 인공생명[2]은 지능을 하나의 독립적인 것으로 보는 기존의 인공지능에 비해 군 지능이 독립적인 것이 아닌 개체간의 상호작용으로부터 이루어진다는 것을 강조하고 있으며 이러한 상호작용으로부터 전역적 행동을 이해할 수 있다고 주장한

다. 자연계에서는 이런 상호작용의 결과로 진화와 창발적 행동이 나타난다. 인공생명 기법을 이용한 많은 연구 결과들이 발표되고 있으며 Mataric[3]은 군 지능을 가진 로봇들이 충돌회피(collision avoidance), 귀향(homing), 확산(dispersion), 집합(aggregation) 등의 기본적인 성질을 가지고 창발적인 행동을 할 수 있음을 보였으며 Kube, Zhang[4]은 전역적 행동을 하는 복수의 행동 기반 로봇을 제어하는 문제를 실험하였다. Craig[5]는 자연계에서 볼 수 있는 새나 어류의 무리 짓기와 충돌 회피를 "boids"라는 프로그램으로 구현했다. 본 논문에서는 자율 이동 로봇의 군 행동을 제어하는 문제를 다룬다. 복수 이동 로봇 에이전트를 이용하여 이들이 다른 개체와 충돌하지 않으면서 무리를 지어 이동하는 복잡한 행동이 단순한 지역적 상호작용에 의해 구현 될 수 있음을 보여주고자 한다. 모든 자율 이동 로봇들은 재구성되는 신경망의 구조로 설계되었으며, 변화하는 환경에 따라서 서로 다른 최적 하중치를 갖도록 되어 있다. 특히 여기서 제안한 군 행동에 대한 제어구조는 기존의 많은 연구 결과들이 복잡하고 연산량이 많았던 것에 비해 단순함을 기본으로 연산량을 줄이는 방향으로 디자인했으며 그 유효성은 컴퓨터 시뮬레이션을 통해 테스트하였다. 컴퓨터 상의 시뮬레이션을 실제 세계에서 완전히 적용하기 어렵다는 것은 주지의 사실이나 이는 진화나 실제 환경의 샘플링 등을 통해 줄일 수 있다는 연구 결과가 있다.[6] 관련된 연구는 군 지능에 관한 창발적인 행동을 실험한 결과들이 있다[6][7].

II. 자율이동로봇(Autonomous Mobile Robot, AMR)

본 논문에서 고려된 자율 이동 로봇은 기본적으로 단일 에이전트이며 이들의 움직임은 각 에이전트들간의 상호작용에 의해 결정된다. 다음은 자율 이동 로봇에 부과된 가정들이다: ① 로봇은 최대 속력과 최소 속력을 가

접수일자 : 1998. 6. 19., 수정완료 : 1998. 12. 18.

민석기 : 중앙대학교 제어계측학과

강 훈 : 중앙대학교 전자전기공학부

※ 본 논문은 한국과학재단(96-0102-13-01-3)에서 지원하여 연구하였습니다.

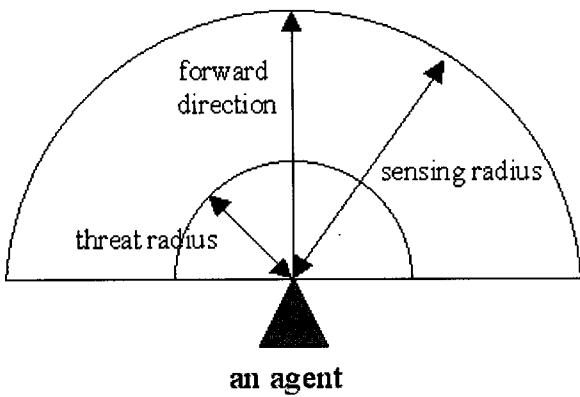


그림 1. 자율 이동 로봇의 센싱 범위.
Fig. 1. Sensor range of an AMR.

지고 있으며 평상시 평균 속도 $|\mathbf{v}^*|$ 을 유지한다. ② 로봇의 회전각은 임의의 각으로 제한되어 있다. ③ 로봇은 자신만의 고유한 상대 좌표를 가지고 있다. ④ 로봇은 센서의 반경이 임의로 정해져 있다. ⑤ 로봇은 자신의 좌표계를 근거로 하여 센서 반경내의 다른 개체들의 거리와 각도를 알 수 있다. 이동 로봇은 센서로부터 입력을 받으며 입력 변수는 다른 로봇이나 장애물의 위치 벡터이다. 센서는 센싱 반경을 가지고 있으며 그 영역은 그림 1과 같다. 즉, 자신의 전방 180° 영역에 들어오는 다른 개체를 감지할 수 있으며 두 가지 영역으로 나누어져 있다. 위협 반경은 로봇의 한 스텝 이동거리보다는 어느 정도 커야 한다. 자율 이동 로봇의 출력변수는 개체가 다음 스텝에서 갈 수 있는 거리와 회전하는 방향, 즉 속도가 된다. 그림 2는 i 번째 이동 로봇의 입력력 변수 구성도를 보여주고 있다. 그림 2에서 $P_{ij}^a(t)$ 와 $P_{ik}^o(t)$ 는 각각 시간 t 에서 i 번째 로봇의 센싱 반경 안에 들어온 j 번째 타 로봇과 k 번째 장애물을 가리키는 위치 벡터이다. 표 1은 각 로봇 에이전트가 가지고 있는 정보를 보여주고 있으며 그 기능은 다음에서 상세히 기술될 것이다.

III. Action network의 구성

그림 2에서 action network의 출력은 로봇의 이동 속도가 되고, 속도는 크기 $|\mathbf{v}_i|$ 와 방향 벡터 \mathbf{d}_i 성분으로 표시할 수 있으므로 action network의 출력은 다음과 같다:

$$\mathbf{v}_i(t) = |\mathbf{v}_i(t)| \cdot \frac{\mathbf{d}_i(t)}{|\mathbf{d}_i(t)|} \quad (1)$$

1. 속력의 결정

속력 $|\mathbf{v}_i(t)|$ 의 결정은 무리를 이루어 다른 로봇과 적절한 거리를 유지하며 진행하도록 정한다. 따라서, 적절한 거리보다 멀다면 가속을 가까우면 감속을 해서 일정한 거리를 유지하도록 배려한다. $\Delta s(t)$ 를 다른 로봇과 적절한 거리를 유지하기 위해 자신이 가감해야할 거리라고 한다면 $\Delta s(t)$ 에 의한 가속도로부터 새로운 속력을 얻을 수 있다:

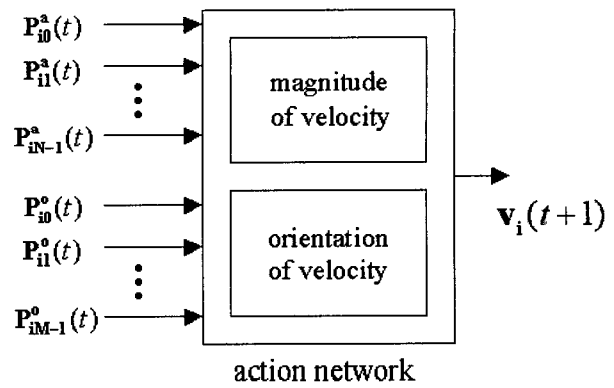


그림 2. 자율 이동 로봇 시스템의 구성도.
Fig. 2. AMR system structure.

표 1. 자율 이동 로봇의 자료 구조.
Table 1. Data structure of an AMR.

자율 이동로봇	자료형	기능
w^a	다른 로봇에 대한 하중값(+)	연결 강도 (로봇의 방향 결정)
w^o	장애물(경계면)에 대한 하중값(+)	연결 강도 (로봇의 방향 결정)
orientation bit	회전방향 설정변수 $\{-1, 0, +1\}$	충돌 회피 모드에서 +1: 오른쪽으로 회전 -1: 왼쪽으로 회전
$\mathbf{v}(r, \theta)$	로봇의 속도 (실수, 실수)	다음 스텝에서 로봇의 위치 결정

$$|\mathbf{a}_i(t)| = \Delta s(t) + |\mathbf{v}^*(t)| - |\mathbf{v}_i(t)| \quad (2-1)$$

$$|\mathbf{v}_i(t+1)| = |\mathbf{v}_i(t)| + |\mathbf{a}_i(t)| \quad (2-2)$$

단, $|\mathbf{v}_{min}| \leq |\mathbf{v}_i(t+1)| \leq |\mathbf{v}_{max}|$

(2-1)에서 $|\mathbf{v}^*|$ 가 들어간 것은 자신이 이동하는 동안 다른 로봇들도 이동하는 것을 고려해준 항이며 각 자율 이동 로봇은 (2-2)에 의해 새로운 속력을 구한다. 로봇은 $\Delta s(t)$ 가 0인 경우(가속이 필요 없는 경우) 평균 속력을 유지한다.

2. 방향 벡터의 결정

방향 벡터를 결정하는 전략은 센싱 영역 안에 어떤 개체가 있느냐에 따라 3가지가 있으며 만약 센싱 범위 안에 다른 개체가 존재하지 않는다면 로봇은 한 스텝 안에서 움직일 수 있는 최대 제한각도 안에서 자유롭게 움직인다. 먼저 우벡터와 좌벡터를 정의하자. 방향 벡터 \mathbf{d}_i 를 기준으로 임의의 벡터 \mathbf{A}_i 가 오른쪽을 향하면 이를 우벡터 $[\mathbf{A}]_+$, 왼쪽을 향하면 좌벡터 $[\mathbf{A}]_-$ 로 정의한다.

2.1 무리 짓기 운동

센싱 영역 안에 다른 로봇이 들어왔을 경우 무리를 만들기 위해서 진행하는 방향을 결정하는 것은 다른 로봇과의 상대적 거리와 방향의 관계임을 알 수 있다. 센싱 영역에 다른 로봇이 들어오면 상대적 거리와 방향 정

보, P_{ij}^a 가 시스템의 입력으로 들어오며 그 입력 위치 벡터에 적절한 하중값 w_i^a 를 곱한 벡터 $w_i^a \cdot P_{ij}^a$ 와 입력 위치 벡터에 수직인 단위 벡터 O_{ij}^a 의 합성 벡터가 다음 스텝에서의 이 로봇의 방향 벡터가 된다. 다른 로봇이 둘 이상 존재한다면 각각에 대해 이런 방향 벡터 성분들을 모두 구하여 합성한 것이 시스템의 출력이 된다. 주 연산자로서 벡터 합을 사용함으로써 시스템의 출력은 다른 로봇들이 밀집한 방향으로 되어 무리를 형성할 확률이 높여준다.

$$d_i(t+1) = \sum_{j=0}^{N-1} (w_j^a(t) \cdot P_{ij}^a(t) + O_{ij}^a(t)) \quad (3)$$

$$O_{ij}^a(t) = \begin{cases} [O_{ij}^a(t)]_i^+ & \text{if } P_{ij}^a(t) = [P_{ij}^a(t)]_i^- \\ [O_{ij}^a(t)]_i^- & \text{if } P_{ij}^a(t) = [P_{ij}^a(t)]_i^+ \end{cases}$$

단, $P_{ij}^a(t) \cdot O_{ij}^a(t) = 0, |O_{ij}^a(t)| = 1$

여기에서 $w_i^a(t)$ 는 로봇에 대한 하중값, $P_{ij}^a(t)$ 는 시간 t 에서 i 번째 로봇의 센싱 반경 안에 들어온 j 번째의 다른 로봇을 가리키는 위치 벡터, $O_{ij}^a(t)$ 는 이에 수직인 단위 벡터이고 N 은 센싱 반경 안에 들어온 로봇의 개수이다. 그림 3은 시간 t 에서 위치 벡터 $P_{ij}^a(t)$, 수직 단위 벡터 $O_{ij}^a(t)$, 하중값 $w_i^a(t)$ 과 방향벡터 $d_i(t)$ 사이의 관계를 보여주고 있으며, 그림 4는 i 번째 로봇의 방향 벡터 $d_i(t)$ 를 결정하는 시스템의 구성도를 나타낸다. 하중값이 커질수록 다른 로봇을 따라가는 속성이 커지며 역으로 작아질수록 피해가는 속성이 커진다. 따라서 적절한 하중값 $w_i^a(t)$ 를 선택하는 것이 중요하다.

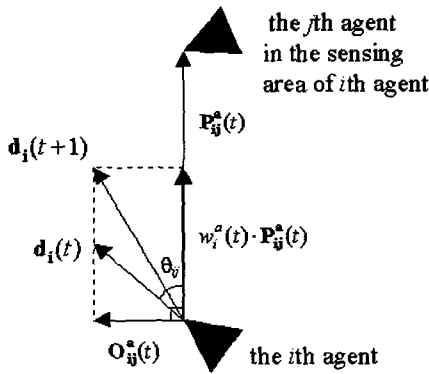


그림 3. $P_{ij}^a, O_{ij}^a, w_i^a$ 와 d_i 의 관계.
Fig. 3. Relations among $P_{ij}^a, O_{ij}^a, w_i^a$ and d_i .

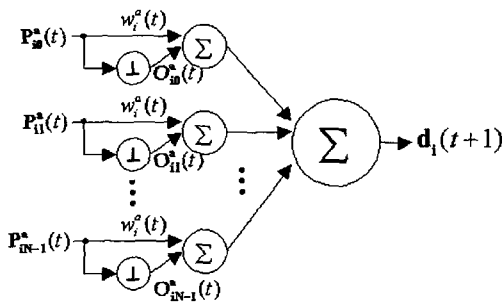


그림 4. i 번째 로봇의 신경 회로망 시스템 구성도.
Fig. 4. Neural networks model of the i th AMR.

2.2 장애물(경계면) 회피 운동

무리 짓기 운동을 하고있지 않고 센싱 영역 내에 장애물(경계면)이 들어왔을 경우의 전략이며 무리 짓기 운동과 유사하나 시스템의 입력으로 자신의 상태, 회전 비트 OB(orientation bit)가 들어가는 것이 다르다. 회전 비트(OB)는 현재 자신이 회전하는 방향을 나타내며 오른쪽으로 회전(OB=1), 왼쪽으로 회전(OB=-1), 장애물 없음 또는 우선권이 높은 장애물의 변경(OB=0)의 3가지 상태가 있다. 임의 로봇이 장애물을 만났을 때, 진행 방향을 결정하는 것은 다음의 식으로 구한다.

$$d_i(t+1) = \min_{j=0}^{M-1} (w_j^o(t) \cdot P_{ij}^o(t) + O_{ij}^o(t)) \quad (4)$$

$$O_{ij}^o(t) = \begin{cases} [O_{ij}^o(t)]_i^+ & \text{if } OB(t) = 1 \\ [O_{ij}^o(t)]_i^- & \text{if } OB(t) = -1 \\ [O_{ij}^o(t)]_i^+ & \text{if } OB(t) = 0 \ \& \ P_{ij}^o(t) = [P_{ij}^o(t)]_i^- \\ [O_{ij}^o(t)]_i^- & \text{if } OB(t) = 0 \ \& \ P_{ij}^o(t) = [P_{ij}^o(t)]_i^+ \end{cases}$$

단, $P_{ij}^o(t) \cdot O_{ij}^o(t) = 0, (|O_{ij}^o(t)| = 1)$

여기에서 $\min_{i=0}^n a_i$ 는 n 개의 벡터 a_i 중에서 최소 크기를 가진 벡터를 반환하는 연산자이며 $w_i^o(t)$ 는 장애물(경계면)에 대한 하중값, $P_{ij}^o(t)$ 는 시간 t 에서 i 번째 로봇의 센싱 반경 안에 들어온 j 번째 장애물을 가리키는 위치 벡터, $O_{ij}^o(t)$ 는 이에 수직인 단위 벡터이며 M 은 센싱 반경 안에 있는 장애물의 개수이다. 로봇은 가장 가까운(우선권이 높은) 장애물(경계면)에 대해 하중값과 위치 벡터에 따라 진행 방향을 결정한다. 장애물(경계면) 회피 운동에서는 무리 짓기 운동과는 달리 회전 비트 OB(t)를 사용한다. 회전 비트가 1과 -1일 경우 수직 단위 벡터는 장애물(경계면)의 위치 벡터가 우벡터인지 좌벡터인지 관계없이 각각 수직 단위 우벡터와 수직 단위 좌벡터로 선택한다. 따라서 다음 스텝에서의 방향 벡터는 회전 비트가 1인 경우 오른쪽으로 -1인 경우 왼쪽으로 향한다. 회전 비트가 0일 경우 수직 단위 벡터는 장애물을 가리키는 위치벡터가 우벡터인지 좌벡터인지에 따라 수직 단위 우벡터나 수직 단위 좌벡터로 선택되며 이에 따른 다음 스텝에서의 로봇의 방향 벡터가 우벡터이면 회전 비트를 1로 좌벡터이면 -1로 설정한다. 회전 비트는 장애물이 센싱 범위에서 없어지거나 가장 가까운(우선권이 높은) 장애물이 변경됐을 경우에 다시 0이 된다. 회전 비트의 영향으로 로봇은 보다 안전하고 효율적으로 장애물을 피한다. 회전 비트는 단지 장애물(경계면) 회피 운동에서만 유효하다. 그림 5는 다음의 알고리즘에 따라 로봇이 장애물을 피하는 과정을 보여주고 있다. 그림 5(b)에서 그림 5(c)사이에서 가장 가까운 장애물이 변경되었으므로 OB(t)가 1에서 0으로 되었다가 다시 장애물의 위치에 따른 $O_{ij}^o(t)$ 의 선택에 의해 -1이 됨을 보였다.

2.3 충돌 회피 운동

위험 반경 안에 다른 로봇이나 장애물이 들어왔을 경

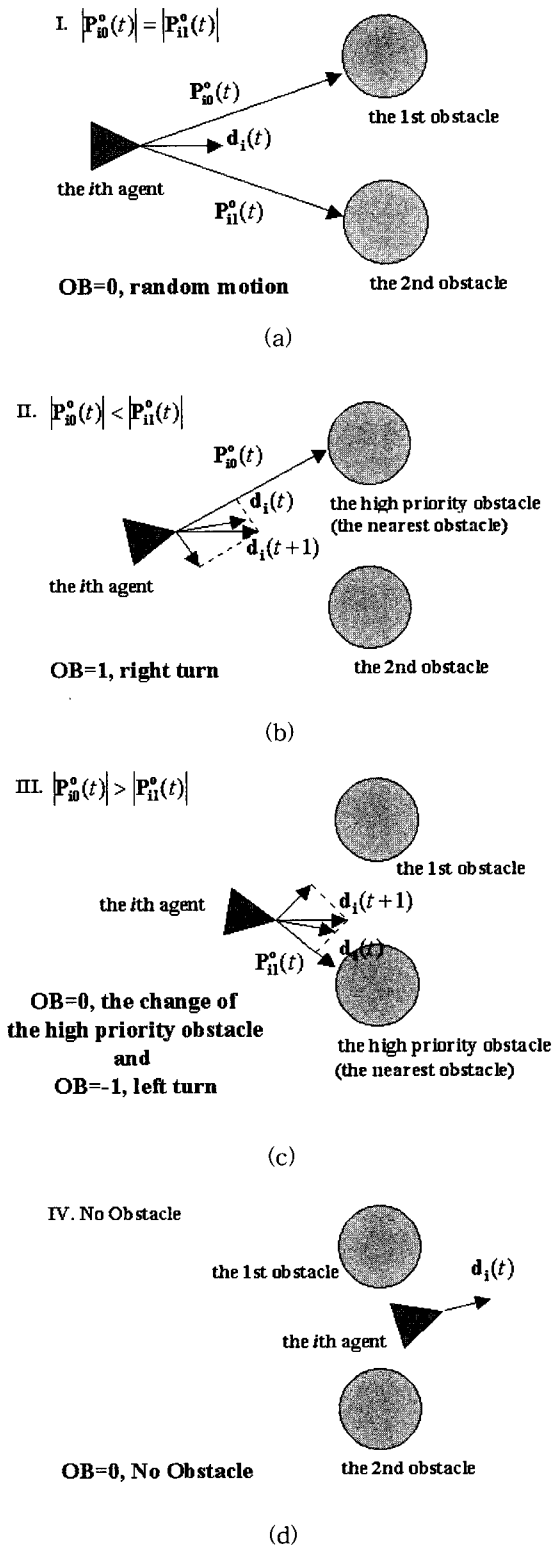


그림 5. 장애물 회피 과정. (a) 로봇의 장애물 인지 (b) 우선권이 높은 장애물의 회피 (c) 장애물의 우선권 변경 (d) 장애물 회피 성공
 Fig. 5. Process of obstacle avoidance. (a) recognition of two obstacles (b) avoidance of the high priority obstacle (c) change of the high priority obstacle (d) success on obstacle avoidance

우, 로봇은 이를 피하는 방향으로 움직여야 하며 가장 우선권이 높은 행동이다. 따라서, 다음과 같이 위험 비율을 정의해서 가장 안전한 방향을 선택하였다.

$$\text{danger ratio} = 1 - |P_{ij}| / TR \quad (5)$$

여기서 TR은 그림 1에서 보여준 위험 반경이며 P_{ij} 는 다른 로봇이나 장애물(경계면)을 가리키는 위치 벡터이다. 그림 6은 위험 비율의 적용 예를 보여주며 위험 영역을 임의의 크기(센서의 개수, n)로 나눈 부분을 세포(cell)라고 부르며 세포 벡터 C로 표현한다. 각 세포는 하나의 진행 방향을 대표하고 있으며 (5)에서 구한 위험 비율 값을 가지고 있다.

가장 작은 값을 가진 세포가 다른 로봇이나 장애물(경계면)과 충돌할 확률이 가장 적은 방향이며 다음 스텝에서의 진행 방향이 된다. 로봇은 한 스텝에서 회전할 수 있는 최대 제한 각도가 있으므로 이를 고려하여 세포를 선택한다.

$$C = [c_1, c_2, \dots, c_n]^T$$

$$d_i(t+1) = \text{DIR} \left(\min_{i=k}^{n-k+1} c_i \right) \quad (6)$$

여기서 DIR(a)는 a 값을 가진 세포가 대표하는 방향을 반환하는 연산자이며 k는 이동 로봇이 한 스텝에서 회전할 수 있는 최대 제한각을 고려한 값이다. 만일 c_i 의 최소값이 복수일 때는 (7)을 두어 번 반복하고 그래도 최소값이 복수이면 그 값 가운데 임의로 하나를 선택한다. (7)을 반복함으로써 자신의 영향이 주위의 세포로 전파되어 높은 값을 가진 세포(위험한 방향) 주위의 위험비율도 높아져 이동 로봇이 더 안전한 방향으로 진행하도록 한다. 본 논문에서는 $V = [1 \ 2 \ 1]$ 로 설정했다.

$$V = [v_1, v_2, \dots, v_m], \quad m \leq n, \quad m \text{은 홀수}$$

$$c_i = \begin{cases} \frac{1}{\sum_{j=1}^m v_j} V \cdot \begin{bmatrix} c_{i-(m-1)/2} \\ \vdots \\ c_{i+(m-1)/2} \end{bmatrix} & \text{if } 1 \leq i \leq n \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

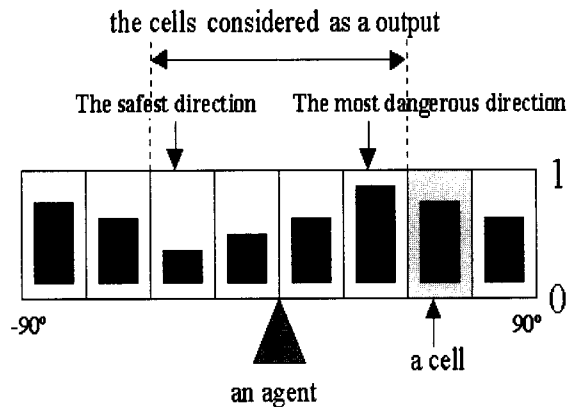


그림 6. 위험 비율의 예.
 Fig. 6. Example of danger ratio.

IV. 자율 이동 로봇의 학습 및 진화

로봇 개체의 주변 환경은 동적으로 변하므로 이에 따라 자신의 행동을 조정해서 원하는 목적을 이루는 것이 효율적이고 능동적이라고 말할 수 있다. 인공생명에서는 이러한 동적인 상호 작용이 중요한 개념으로 복잡 적응계를 나타낸다. 따라서 인공 생명에 의한 자율 이동 로봇 시스템의 구조는 궁극적으로 환경과 시스템의 상호 작용을 통해 원하는 목표를 성취할 수 있어야 한다. 또한 문제를 수행함에 있어서 사전에 명백히 프로그래밍 되어있지 않더라도 예측하지 못한 문제가 발생했을 경우라도 이에 적응 및 학습을 통해 대처해 나가는 능력이 있어야 한다. 근래에 적응 행동과 진화 연산을 구현하는 기법으로 생물체의 적응 메커니즘을 인공적으로 적용한 구조가 관심을 끌고 있으며 신경 회로망, 강화 학습, 유전자 알고리즘, 유전자 프로그래밍, 진화 전략, 진화 프로그래밍, 면역 시스템 등의 장점을 융합한 모델이 많이 연구되고 있다. 여기서는 신경 회로망[8][9]과 강화 학습[10]을 혼합하여 변화하는 환경에 적응하는 모델을 구현하였다. 신경 회로망은 동물의 신경 세포를 기반으로 한 구조이며 강인한 학습 능력을 가지고 있고, 강화 학습은 환경에 대한 사전 지식이 없는 경우 시행 착오를 거쳐 행동의 학습을 통해 상태와 행동을 연결시켜주는 학습 구조로 Sutton의 adaptive heuristic critic구조[11]와 Watkins의 Q-학습법[12] 등이 있다. 본 연구에서는 교사 학습에 의한 진화의 예로 자율 이동 로봇군이 특정한 편대의 모양을 유지하며 무리를 지어 이동하는 것을 목표로 설정하였으며 학습을 통해 하중값을 조정함으로써 원하는 목적을 달성하도록 하였다. 편대의 모양은 그림 7과 같으며 각 로봇은 자신의 선두에 대해 일정한 거리 z 와 각도 ϕ 를 유지하며 이동한다. 선두는 특별히 정해져 있지 않으며 임의의 로봇이 될 수 있고 언제든지 변할 수 있다. 선두는 가장 근거리의 로봇으로 선택한다.

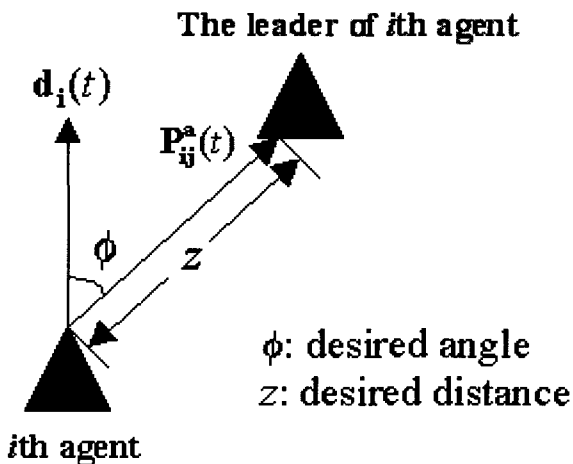


그림 7. 원하는 편대의 부분적 형상.
Fig. 7. Partial pattern of the desired formation.

1) 거리 z 의 보정 :
속력을 조정하여 원하는 거리를 유지하도록 한다. 아

래와 같이 $\Delta s(t)$ 를 정의하고 (2)에 의해서 새로운 속력을 얻는다.

$$\Delta s(t) = \begin{cases} -TR & \text{if } 0 \leq z \leq TR \\ 0 & \text{if } z - a \leq |P_{ij}^a(t)| \leq z + a \\ |P_{ij}^a(t)| - z & \text{otherwise} \end{cases} \quad (8)$$

단, $0 \leq z \leq SR$

여기서 a 는 임의의 상수이며 SR과 TR은 각각 그림 1에서의 센싱 반경과 센싱 반경내의 위협반경이다.

2) 각도 ϕ 의 보정 :

선두와 원하는 각을 유지하도록 하중값 w_i^c 를 조정한다. 하중값 수정 규칙은 (9-1)과 같이 제곱 오차를 정의함으로써 일반적으로 사용되는 오차 최소화 방법에 따른다:

$$E_i(t) = \frac{1}{2} \epsilon_i(t)^2, \epsilon_i(t) = (\phi - \theta_{ii}(t)) \quad (9-1)$$

$$w_i^c(t+1) = w_i^c(t) - \alpha \cdot \epsilon_i(t) \quad (9-2)$$

여기서 θ_{ii} 은 자신의 선두와의 각도 차이이다.

3) 강화 신호에 의한 전략 :

상태에 따른 보상과 벌칙 신호를 주어 로봇이 보다 효율적인 방향으로 원하는 목적을 이루도록 하였다. 강화 신호 $r_i(t)$ 평가함수와 그것에 의한 하중값 수정규칙은 다음과 같다:

$$r_i(t) = \begin{cases} 1 & \text{if } \Delta s(t) < 0 \\ 0 & \text{if } \Delta s(t) = 0 \\ -1 & \text{if } \Delta s(t) > 0 \end{cases} \quad (10-1)$$

$$w_i^c(t+1) = w_i^c(t) - \alpha \cdot r_i(t) \quad (10-2)$$

그림 8은 강화 신호가 있을 경우 빠르고 안전하게 선두를 따라가면서 원하는 무리 형성을 이루는 것을 보여준다. 따라서, 하중값의 수정 규칙은 (9-2)와 (10-2)에서 다음과 같이 정리된다.

$$w_i^c(t+1) = w_i^c(t) - \alpha(r_i(t) + \lambda \epsilon_i(t)) \quad (11)$$

여기서, α 는 학습률, λ 는 비례상수이다. 그림 9는 신경 회로망의 학습 구조를 사용한 자율 이동 로봇 개체의 구성도이며, 다음의 알고리즘은 행동-학습의 단계를 순차적으로 설명해주고 있다.

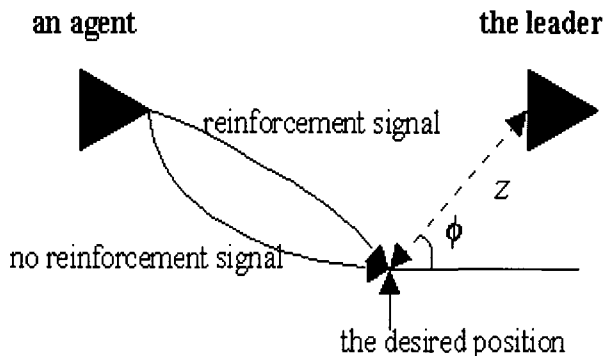


그림 8. 강화 신호의 영향.
Fig. 8. Effect of reinforcement signal.

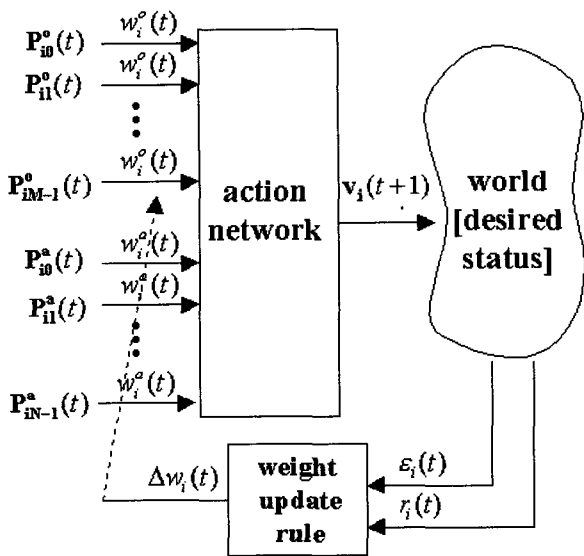


그림 9. 자율 이동 로봇의 행동-학습 구조.
Fig. 9. Action-learning structure of an AMR.

- ※ 행동-학습의 알고리즘:
1. 현재의 상태를 인식한다.
 2. action network을 통해 행동값을 얻는다.
 3. 행동을 수행한다.
 4. 행동에 따른 새로운 상태를 인식한다.
 5. 새로운 상태에 따른 강화 신호 r_i 와 오차 ϵ_i 를 얻는다.
 6. 오차와 강화 신호에 따른 Δw_i^o 값에 따라 action network의 하중값 w_i^o 을 수정한다.
 7. 단계 2로 되돌아간다.

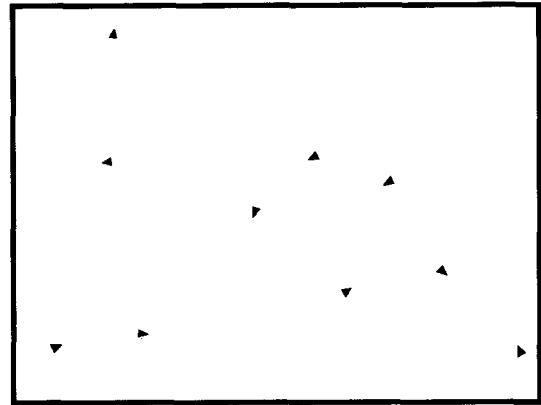
V. 모의 실험에 의한 검증

1. 모의 실험 결과

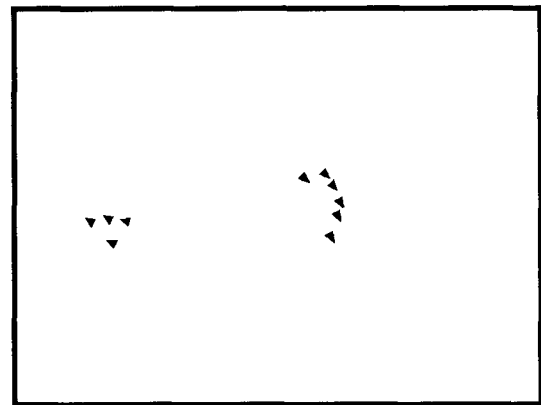
위의 행동-학습에 따른 시뮬레이션을 수행하여 여러 가지 일치하는 결과를 얻을 수 있었다. 아래의 모든 모의 실험에서 센싱 반경 100, 위협 반경 20, 평균 속도 2의 시뮬레이션 환경은 공통적으로 설정했다. 그림 10은 10개의 로봇이 학습을 사용하지 않고 고정된 하중값을 사용했을 경우 무리 지어 이동하는 것을 보였다. 이 경우 타 로봇에 대한 하중값은 0.04로 정하였다. 그림 11은 하나의 로봇이 25개의 장애물과 경계면을 피해 움직이는 궤적을 보였으며 그림 12는 8개의 로봇들이 무리를 지으면서 장애물을 피하는 것을 보여주고 있다. 이 경우에는 장애물의 개수 10개, 타 로봇에 대한 하중값 0.04, 장애물에 대한 하중값 0.03으로 정하였다. 그림 13은 6개의 로봇들이 10개의 장애물과 경계면을 피하면서 특정한 형태로 무리지어 이동하는 과정을 순차적으로 보여주고 있다. 시뮬레이션 환경은 원하는 거리 $z=40$, 원하는 각도 $\phi=30^\circ$, 평균 속력 2, 최고 속력 4, 최저속력 1, $\alpha=0.001$, $\lambda=1$, $a=6$, 장애물의 하중값은 0.03으로 정하였다.

그림 14는 5개의 로봇이 원하는 거리 z 를 40으로 고정시키고 원하는 각도 ϕ 를 변화시켰을 때 나타나는 편대

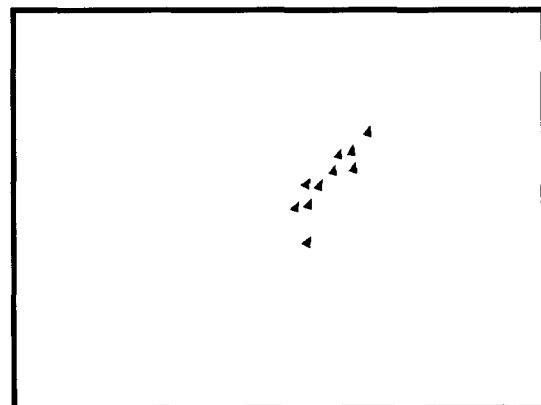
의 모습이다. 그림 15는 하나의 로봇이 동적인 2개의 장애물과 위아래의 경계면에 대해서 충돌을 회피하는 과정을 나타낸 것이며 장애물 옆의 화살표는 장애물의 진행 방향을 나타낸다. 이 경우 장애물에 대한 하중값은 0.01로 설정했다.



(a)



(b)



(c)

그림 10. 고정된 하중값에 대한 무리짓기 이동모습. (a) 초기 상태, (b) 두 개의 무리로 이동, (c) 하나의 무리로 이동.

Fig. 10. Flocking behavior of AMRs with fixed weight. (a) initial status, (b) movement in two groups, (c) movement in a group.

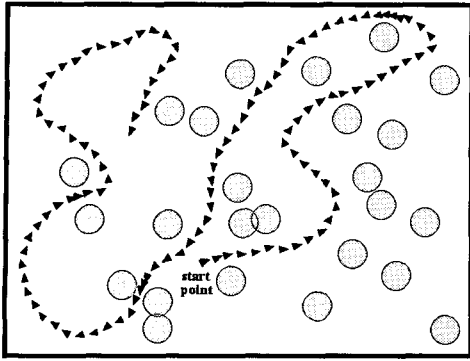


그림 11. 로봇의 장애물 회피 궤적.
Fig. 11. Trajectory of an AMR to avoid obstacles.

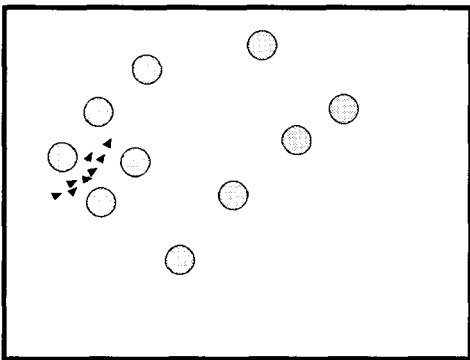
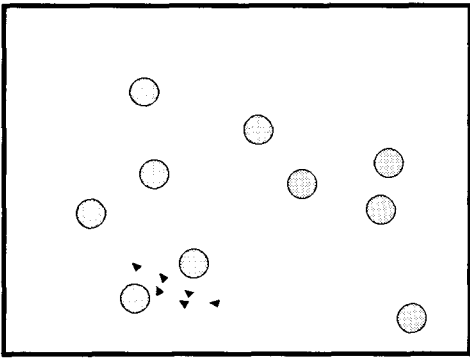
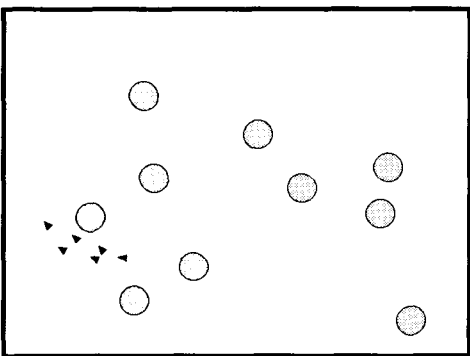


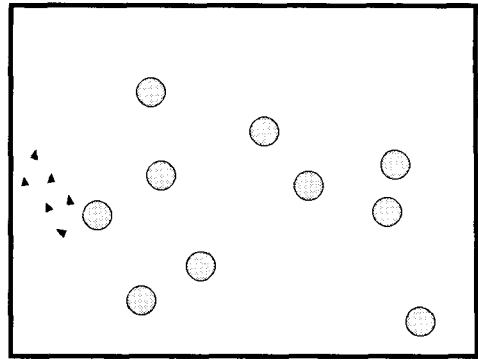
그림 12. 로봇의 무리 짓기와 장애물 회피.
Fig. 12. Flocking and obstacle avoidance of AMRs.



(a)



(b)



(c)

그림 13. 학습 규칙을 사용한 자율 이동 로봇의 장애물(경계면) 회피 및 무리짓기 이동과정.
Fig. 13. Process of obstacles avoidance and the flocking of AMRs using learning rule.

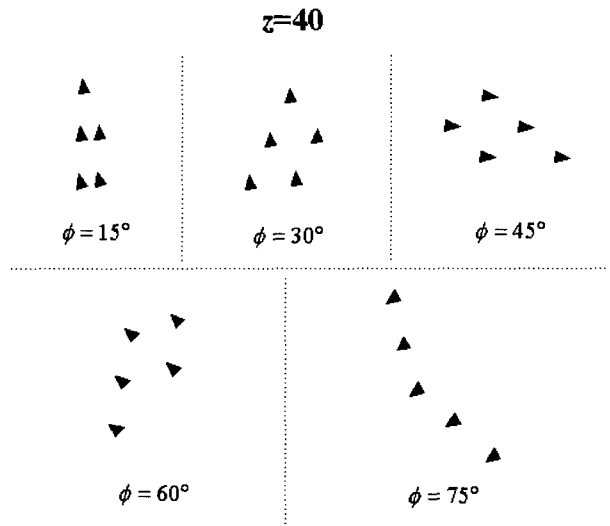


그림 14. 원하는 각도에 따른 편대의 형태.
Fig. 14. Patterns of the group according to ϕ .

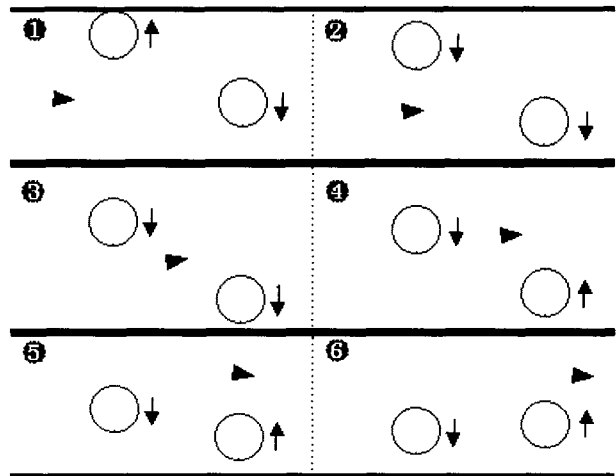


그림 15. 움직이는 장애물의 회피.
Fig. 15. Avoidance of the moving obstacles.

2. 성능 지수와 자율 이동 로봇군의 학습에 의한 진화

그림 16은 자신의 선두에 대해 일정한 위치를 유지하게 해주는 최적의 하중값 w_i^* 이 존재함을 보여준다. 이 값은 원하는 위치에서 최적의 값이며 다른 위치에서는 최적의 값이 아니다. 이 최적의 하중값은 다음과 같다.

$$w_i^* = \frac{1}{z \cdot \tan \phi}, z > TR \quad (12)$$

여기서 TR은 그림 1의 위협 반경이다. 따라서, 원하는 편대 형태의 무리가 형성된다면 선두를 제외한 로봇들은 최적의 하중값 w_i^* 으로 수렴할 것이며 하중값 궤적 폭선을 조사하여 시스템의 성능 지수를 확인할 수 있다. 그림 17은 위협 반경을 20으로 설정하였을 때 원하는 거리 z 및 원하는 각도 ϕ 대 최적의 하중값 w_i^* 에 대한 그래프를 보여주고 있다. 대부분 0.5이하의 작은 값을 가지고 있지만 z 와 ϕ 가 작은 경우, 즉 가까운 거리를 유지하면서 선두를 일직선으로 뒤따라가기 위해서는 하중값 w_i^* 이 급격히 증가해야 한다는 것을 알 수 있다.

그림 18(a)는 6개의 로봇이 원하는 거리를 40, 원하는 각도를 25°로 설정하고 모의 실험하였을 때 시간에 따른 각 로봇의 하중값 w_i^a 궤적을 보여주고 있으며, 그림 18(b)는 시간 700~800사이의 하중값 w_i^a 궤적을 확대한

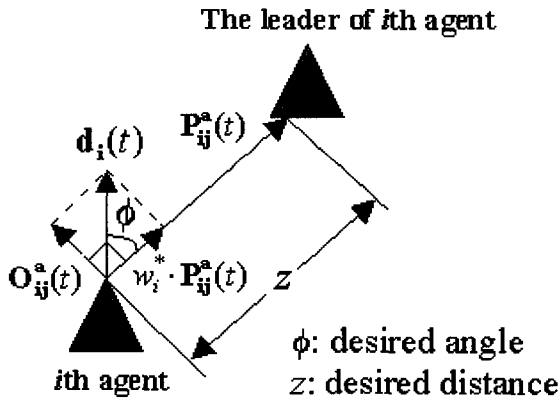


그림 16. w_i^* 와 d_i 의 관계.
Fig. 16. Relations between w_i^* and d_i .

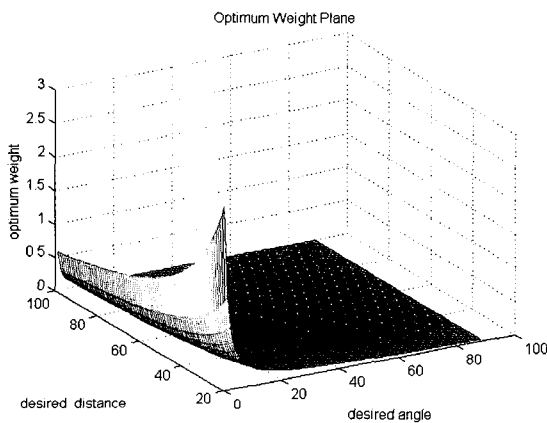
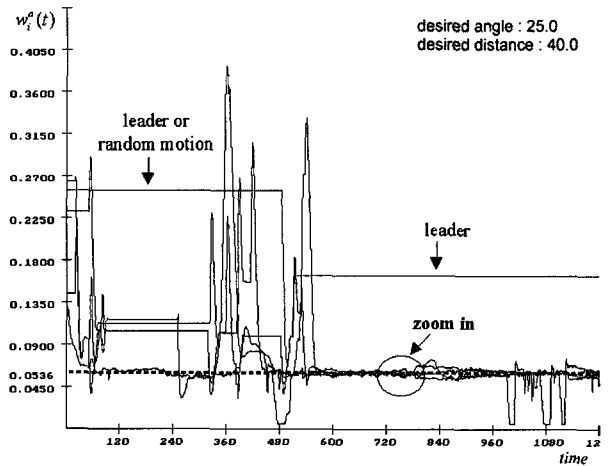


그림 17. 최적의 하중값 곡면.
Fig. 17. Optimum weight plane.

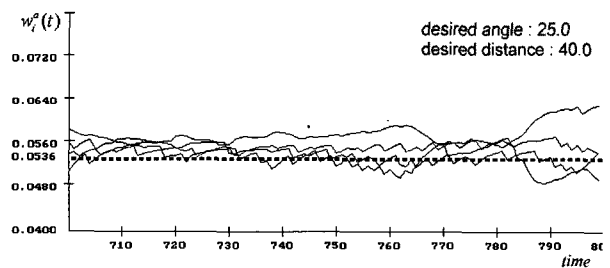
그림이다. 그림 18(a)에서 시간 550이후 완전한 하나의 무리를 형성했음을 알 수 있다. 최적의 하중값 w^* 은 $0.0536(1/40 \tan(25^\circ))$ 이며 그림 18(b)에서 알 수 있듯이 선두를 따라가는 로봇들의 하중값은 선두가 무작위 움직임을 하기 때문에 최적의 하중값 근처에서 진동함을 알 수 있다. 그림 19(a)에서는 비교적 적은 수의 로봇에 대한 하중값 궤적을 보였고, 그림 19(b)는 이에 대한 절대 오차를 표시한 그래프이다.

VI. 결론

자연계에서 새나 어류가 무리지어 다니는 행동을 복수의 자율 이동 로봇을 통해 구현하였으며, 개체간의 지역적 상호작용으로부터 전역적 균행동이 유발되는 것을 살펴보았다. 제안된 자율 이동 로봇의 구조는 신경 회로망의 학습과 강화 신호를 사용하여 인공 생명의 효율적인 전략을 가지고 환경에 따라 적응 행동을 이루는 것을 보여준다. 특히 제안된 신경 회로망은 단순성을 기초로 분산된 시스템으로 구현되었으며 자율 이동 로봇의 행동은 주어진 장애물 환경, 즉 정적 환경 뿐만 아니라 동적 환경에서 움직이는 다른 개체에 대해서도 적응하여 학습을 통한 최적의 하중값을 스스로 선택해나가는 것을 확인하였다.

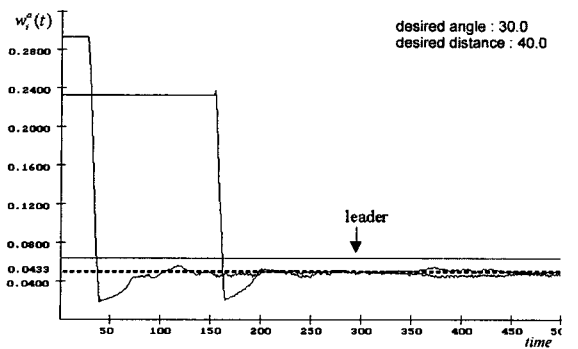


(a)

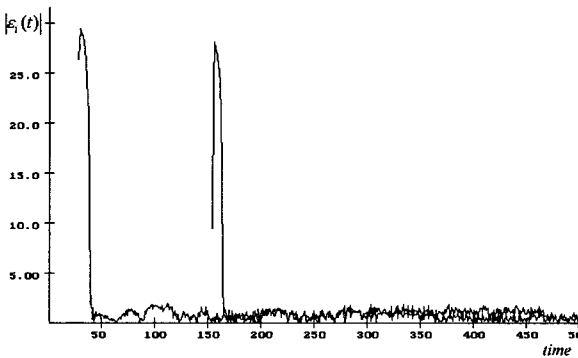


(b)

그림 18 (a) 자율 이동 로봇의 하중값 궤적 I, (b) 최적의 하중값 근처에서의 진동.
Fig. 18 (a) Weight trajectory of the AMRs I, (b) Oscillation of weight nearby the optimum weight.



(a)



(b)

그림 19 (a) 자율 이동 로봇의 하중값 궤적 II,
(b) 절대 오차 $|e_i|$ 궤적.

Fig. 19 (a) Weight trajectory of the AMRs II,
(b) Absolute error trajectory.

참고문헌

[1] R. A. Brooks, "Behavior humanoid robotics," *Proc. of Int. Conf. on IROS*, pp. 1-8, 1996.

[2] C. Langton, "Artificial life" in *Artificial Life*, C. Langton (ed.), Addison-Wesley, pp. 1-47, 1989.
 [3] M. J. Mataric, "Designing emergent behaviors: from local interactions to collective intelligence," *Proc. 2nd Int. Conf. on Simulation of Adaptive Behavior*, pp. 432-441, 1993.
 [4] C. R. Kube and H. Zhang, "Collective robotic intelligence," *Proc. 2nd Int. Conf. on Simulation of Adaptive Behavior*, pp. 460-468, 1993.
 [5] C. W. Reynolds, "Flocks, herds, and schools: a distributed behavioral model," *Proc. of SIGGRAPH*, pp. 25-34, 1987.
 [6] O. Miglino, H. H. Lund and S. Nolfi, "Evolving mobile robots in simulated and real environments," *Artificial Life*, vol. 2, no. 4, pp. 417-434, MIT Press, 1996.
 [7] 민석기, 제갈욱, 강훈, "자율 이동 로봇군의 생명체 행동," 한국 퍼지 및 지능 시스템 학회(KFIS), 97년도 추계학술대회 논문집, vol. 7, no. 2, pp. 83-86, 1997.
 [8] J. A. Freeman and D. M. Skapura, *Neural Networks: Algorithms, Applications, and Programming Techniques*, Addison-Wesley, 1991.
 [9] M. T. Hagan and H. B. Demuth, *Mark Beale, Neural Network Design*, PWS Publishing, 1996.
 [10] J. S. R. Jang, C. T. Sun, and E. Mizutani, *Learning from Reinforcement, Neuro-Fuzzy and Soft Computing*, pp. 258-300, Prentice Hall, 1997.
 [11] R. S. Sutton, "Learning to Predict by the Methods of Temporal Differences," *Machine Learning*, vol. 8, pp. 9-44, 1988.
 [12] C. J. C. H. Watkins and P. Dayan, "Technical Note : Q-Learning," *Machine Learning*, vol. 8, pp. 279-292, 1992.



민 석 기

1996년 중앙대 제어계측공학과 졸업, 1996년-현재 동대학원 제어계측학과 석사과정, 관심분야는 신경회로망, 퍼지시스템 및 제어, 인공지능, 진화연산.



강 훈

1982년 서울대 전자공학과 졸업, 동대학원 석사(1984), GEORGIA INST. OF TECH. 공학박사(1989) 1992년 3월-현재 중앙대학교 전자전기공학부 부교수 1997년 관심분야는 퍼지시스템 및 제어, 신경회로망 및 패턴인식, 인공지능, 진화연산, 유전 프로그래밍, 로보틱스 및 지능제어.