

클러스터링에 의한 자율 이동 로봇의 장애물 회피 알고리즘

Implementation of the Obstacle Avoidance Algorithm of Autonomous Mobile Robots by Clustering

김 장 현*, 공 성 곤

송실대학교 전기공학과

Jang-Hyun Kim*, Seong-Gon Kong

Department of Electrical Engineering, Soongsil University

Abstract

In this paper, Fundamental rules governing group intelligence "obstacle avoidance" behavior of multiple autonomous mobile robots are represented by a small number of fuzzy rules. Complex lifelike behavior is considered as local interactions between simple individuals under small number of fundamental rules. The fuzzy rules for obstacle avoidance are generated from clustering the input-output data obtained from the obstacle avoidance algorithm. Simulation shows the fuzzy rules successfully realizes fundamental rules of the obstacle avoidance behavior.

1. 서 론

인공생명(artificial life)의 연구에서는 모든 생명현상을 나타내고 있는 개체는 기능이 단순한 여러 개의 구성요소들에 의하여 조직되어 있으며, 그러한 구성요소들이 기본적인 규칙에 의하여 서로 국부적으로 상호작용을 함으로써 생명현상을 특징짓는 복잡한 행동패턴들을 나타내고 있다고 가정한다. 이것은 각 구성요소의 개별행동을 지배하는 전체적인 제어 메커니즘 없이 많은 수의 단순한 구성요소들의 조직화된 상호작용에 의하여 복잡한 생명현상을 보일 수 있다는 것을 의미한다[1][2].

인공생명 분야에서 퍼지시스템 이론을 도입하려는 시도는 아직 초기라고 할 수 있다. 현재

인공생명을 구현하는데 있어서 진화연산이나 셀룰러 오토마타, 신경회로망과 같은 방법들을 주로 사용하고 있다. 이러한 방법들은 학습이나 자연의 선택과정으로부터 자기구성할 수 있는 특징이 있으므로 “밑에서 위로(bottom up)”의 처리과정을 택하고 있는 인공생명의 접근방식에 적합하다. 최근 퍼지 시스템 연구의 경향은 전문가의 지식체계를 직접 적용하여 퍼지 시스템을 구성하는 것으로부터 적용 알고리즘에 의해 데이터의 형태로 주어지는 지식을 학습하여 퍼지 시스템을 모델링하는 것으로 바뀌어가고 있다. 이러한 연구는 신경회로망이나 유전 알고리즘 기법들의 장점을 선택적으로 결합하는 하

이브리드 방식의 퍼지 시스템을 의미한다.[3][4]

이 논문에서는 생명현상을 나타내고 있는 개체가 다수의 단순한 구성요소들의 집합으로 이루어져 있으며 그러한 구성요소들이 가지고 있는 소수의 기본규칙들에 의하여 서로 국부적인 상호작용을 함으로써 복잡한 행동패턴들을 나타낸다는 가정 아래, 여러 대의 자율이동로봇(autonomous mobile robot)들의 군지능을 나타낼 수 있는 적은 수의 기본규칙을 찾아내고 이 규칙들을 클러스터링하여 IF-THEN 형태의 퍼지규칙을 생성하고 국부적 상호작용을 표현한다. 각 자율이동 로봇은 기능이 매우 제한되어 있으며, 개체간 명확한 형태의 통신을 할 수 없음에도 불구하고, 한 개체가 서로 다른 개체와의 거리와 위치에 의해서 서로 충돌하지 않고 회피하는 자율이동로봇의 알고리즘을 생각해보았다. 이러한 자율이동로봇의 장애물 회피 알고리즘의 특성을 몇 개의 규칙으로 표현하기 위해 장애물 회피 알고리즘으로 정확하게 계산된 자율이동로봇의 출력력 데이터를 클러스터링하여 적은 양의 퍼지규칙을 갖는 자율이동로봇으로 모델링하였다.[7]

2. 장애물 회피 알고리즘

2.1 자율이동로봇의 장애물 회피

자율이동 로봇은 스스로의 판단아래 자유로이 이동할 수 있는 독립적인 개체이지만, 각 개체의 기능은 제한되어 있으며 다른 개체와 명확한 형태의 통신도 불가능하다. 복잡한 생명현상을 특징으로 하고 있는 인공생명 시스템을 구현하기 위하여, 여러 대의 자율이동로봇을 설계하였으며 방향성 및 속도를 갖고 있는 자율이동로봇이 다른 자율이동로봇 또는 장애물과의 충돌을 회피하는 군행동(group behavior)을 정의하였다. 자율이동로봇의 장애물 회피 알고리즘에 의한 군행동은 12개의 센서에 의해 장애물의 인식을 판별하고 다른 자율이동로봇도 센서에 인식되었을 때 장애물로 인식하여 회피하게 된다. 센서 입력 시 둘 이상의 센서에 입

력을 받았을 때 두 센서로 입력된 장애물의 거리를 비교하여 가장 가까운 장애물을 회피하게된다.

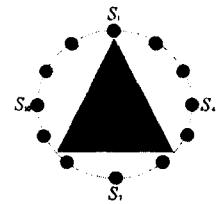


그림 1: 자율이동로봇의 모델

그림 1은 이 알고리즘에 이용된 자율이동로봇의 모델이며 둑근 타원은 자율이동로봇을 나타내며 중앙에 삼각형은 자율이동로봇의 방향성을 표현하기 위하여 나타내었다. 자율이동로봇의 장애물 회피 알고리즘은 먼저 다른 자율이동로봇과 장애물의 위치를 12개의 센서를 이용하여 파악하게 된다.

12개의 센서(S_1, S_2, \dots, S_{12})는 자율이동로봇의 방향성을 기준으로 0° 로 시작하여 시계방향으로 30° 씩 위치하고 있다. 자율이동로봇의 센서의 각도는 식(1)로 표현된다.

$$S_i = \left(\frac{360}{p} \right) \times i \quad (1)$$

p 는 센서의 개수를 의미하며 i ($i=1, 2, \dots, p$)는 로봇에 위치한 센서의 번호를 의미한다. 자율이동로봇은 센서에 다른 자율이동로봇이나 장애물을 인식하지 않았을 경우 자신이 가지고 있는 행동패턴으로 이동을 하게 되며 다른 자율이동로봇이나 장애물이 근접하여 센서에 인식하게 되면 어느 센서에 감지되었나를 확인하고 거리를 계산하게 된다. 그 다음 자율이동로봇은 다른 로봇과 장애물을 회피하여 이동을 행하기 위해 자율이동로봇이 이동해야 할 상대

적인 거리와 회전해야 할 각도를 계산하며 자율이동로봇은 실제적으로 제한된 거리 \hat{r}^{k+1} 와 각도 $\hat{\theta}^{k+1}$ 만으로 이동할 수 있다.

자율이동로봇의 장애물 회피를 위한 이동을 그림 2에 나타내었다. 자율이동로봇 R_1 이 일정한 움직임을 보이다가 다른 자율이동로봇이나 물체가 충돌범위 안에 존재하여 센서로 인식하게 되면 장애물로 인식하고 상대적인 각도를 계산하여 장애물을 회피하여 이동하게 된다.

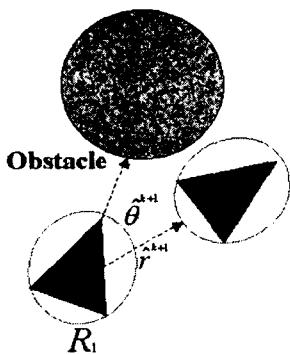


그림 2: 자율이동로봇의 장애물 회피

자율이동로봇은 이동 후 다시 자신의 방향성으로 이동을 하며 이러한 과정을 반복하게 된다.

2.2 자율이동로봇의 퍼지 시스템

자율이동로봇의 장애물 회피 알고리즘 입출력 변수는 먼저 자율이동로봇이 다른 자율이동로봇이나 장애물과의 충돌 회피시 물체를 감지한 센서의 각도와 로봇과 장애물과의 센서에 입력된 거리와 로봇이 가지고 있는 방향성이며 출력으로는 로봇이 회전할 수 있는 상대적인 각도와 로봇이 이동할 수 있는 상대적인 거리를 이용하였다. 입출력 변수의 구성도는 그림 3에 나타내었다.

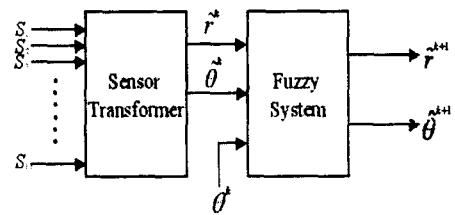


그림 3: 장애물 회피 퍼지 시스템

그림 3은 자율이동로봇이 장애물을 회피할 때의 입출력 변수의 구성으로서 k번째 단계에서 로봇의 센서에 물체가 인식되는지의 여부를 확인하고 자율이동로봇의 센서로 물체를 인식하였다면 개체의 충돌반경안에 물체의 존재 유무를 파악하게 된다. 만약 물체가 로봇의 충돌반경안에 존재하지 않으면 자신이 가진 방향성으로 이동을 하게 되고 물체가 충돌반경안에 존재하게 되면 물체를 장애물로 인식하고 회피하게 된다.

퍼지 시스템은 자율이동로봇과 장애물과의 거리 \hat{r}^k 와 장애물을 인식할 때의 센서의 각도 $\hat{\theta}^k$ 와 자기 자신의 방향성 θ^k 를 입력으로 실제 이동하여야 할 거리 \hat{r}^{k+1} 와 실제 회전하여야 할 각도 $\hat{\theta}^{k+1}$ 를 계산한다. 이와 같이 주어지는 자율이동로봇은 입출력 변수를 가지고 입출력 데이터를 얻어 클러스터링을 하여 장애물 회피를 위한 퍼지규칙을 얻었다.

3. 퍼지 규칙 생성

3.1 클러스터링 알고리즘

클러스터링을 이용한 퍼지 시스템 모델링에는 Sugeno 퍼지모델이 사용되며, 모델링 과정은 주어진 입출력 공간에서의 데이터를 클러스터링하여 클러스터 중심을 결정하는 과정과 결정된 클러스터 중심에 기초하여 퍼지규칙으로 매핑하는 과정으로 나누어진다. Sugeno 퍼지모델의 규칙 형태는 MISO인 경우, 식 (2)처럼 주어진다.

$$\text{IF } x_1 \text{ is } A_1, x_2 \text{ is } A_2, \dots, x_m \text{ is } A_m \quad \text{THEN } y_i = a_0 + a_1 x_1 + \dots + a_m x_m \quad (2)$$

여기서 A_1, A_2, \dots, A_m 는 퍼지 입력 멤버쉽 함수로서 가우시안 함수를 사용하였으며, a_0, a_1, \dots, a_m 는 데이터가 주어지면 결정되어야 하는 상수이다. 클러스터링을 통해 클러스터 중심을 결정하는 과정은, 주어진 입력 데이터를 $[0,1]$ 로 정규화 시키고 각 데이터에서 포텐셜값을 식 (3)로 계산한다.

$$P_i = \sum_{m=1}^M \exp(-\alpha \|x_i - x_m\|^2), \quad \alpha = 4/r_a^2 \quad (3)$$

이때 $i=1, \dots, M$ 이며 r_a 는 양의 상수로서 r_a 밖의 데이터는 포텐셜 값에 영향을 거의 주지 못한다. 첫 번째 클러스터 중심 x_1^* 은 P_i 중에서 가장 큰 포텐셜값 P_1^* 을 갖는 위치의 데이터가 된다. 두 번째 클러스터 중심 x_2^* 의 결정은 첫 번째 클러스터 중심의 영향을 제거한 포텐셜 값을 식 (4)과 같이 구하며 이때 가장 큰 포텐셜 값 P_2^* 의 위치가 두 번째 클러스터 중심이 된다.

$$P_i^* = P_i - P_1^* \exp(-\beta \|x_i - x_1^*\|^2), \quad \beta = 4/r_b^2 \quad (4)$$

이때 r_b 는 r_a 보다 큰 양의 상수로서 결정된 클러스터 중심 근처에 클러스터 중심이 생성되는 것을 막는다. m 번째 클러스터 중심 x_m^* 은 $m-1$ 번째 클러스터 중심의 영향을 식 (5)와 같이 제거한 후 계산한 포텐셜 값 중 최대인 위치의 데이터가 된다.

$$P_i^* = P_i - P_{m-1}^* \exp(-\beta \|x_i - x_{m-1}^*\|^2) \quad (5)$$

이러한 과정은 포텐셜 값이 일정한 범위 ($\bar{\varepsilon}, \underline{\varepsilon}$)에 도달할 때까지 반복하게 된다.

M 개의 클러스터 중심 $\{x_1^*, x_2^*, \dots, x_M^*\}$ 이 결정되면 M 개의 퍼지 규칙으로 매핑하게 된다. 퍼지

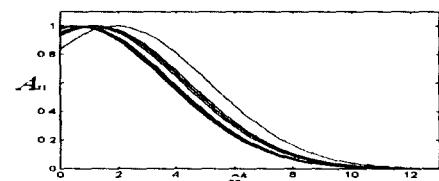
규칙의 전건부 파라미터 중 입력 멤버쉽함수의 위치는 클러스터 중심의 입력공간의 좌표로, 폭은 경험에 기초하여 결정하며 후건부 파라미터 (a_0, a_1, \dots, a_m)는 선형 최소 자승법을 이용해서 최적화 한다.

3.2 퍼지 규칙의 생성

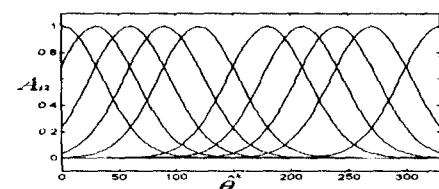
클러스터링을 위한 입출력 데이터는 150번씩 20회를 이동하면서 랜덤하게 생성된 4개의 로봇의 이동으로부터 추출하였으며 클러스터링에 사용 된 파라미터로는 $\gamma_a=0.5$, $\gamma_b=2.5$, $\bar{\varepsilon}=0.3$, $\underline{\varepsilon}=0.1$ 과 같다. 클러스터링 결과 자율이동로봇이 장애물 회피 시 16개의 퍼지규칙을 생성하였으며 규칙을 가지는 입력 멤버쉽 함수를 그림 4에 나타내었다.

IF \hat{r}^k is A_1 , $\hat{\theta}^k$ is A_2 , θ^k is A_3

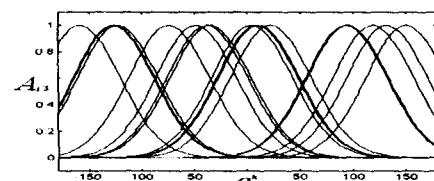
$$\text{THEN } \hat{r}_i^{k+1} = a_0 + a_1 \hat{r}^k + a_2 \hat{\theta}^k + a_3 \theta^k \\ \hat{\theta}_i^{k+1} = b_0 + b_1 \hat{r}^k + b_2 \hat{\theta}^k + b_3 \theta^k \\ (i = 1, 2, 3, \dots, 16)$$



(a)장애물과의 거리(\hat{r}^k)



(b)센서의 각도($\hat{\theta}^k$)



(c) 로봇이 향한 방향($\hat{\theta}^{k+1}$)

그림 4: 퍼지규칙의 입력멤버쉽함수

그림 4는 입력 멤버쉽함수의 형태를 표현하였고, r^k 는 장애물과의 거리를 나타내고 θ^k 는 자율이동로봇의 센서의 각도를 나타내며, θ^k 는 자율이동로봇 자신이 향한 방향을 나타낸다. 각 입력 멤버쉽함수 폭은 r^k 의 경우 3.18, θ^k 의 경우 35.00, 그리고 θ^k 의 경우 38.17으로 고정하였다.

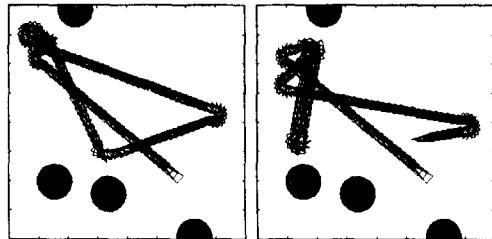
표 1: 출력 변수의 파라미터

제수 구칙	a_{0i}	a_{1i}	a_{2i}	a_{3i}	b_{0i}	b_{1i}	b_{2i}	b_{3i}
1	0.3628	-0.0005	-0.0018	-0.0358	-0.0273	0.4025	1.6860	7.3220
2	0.0164	-0.0003	0.0006	-0.0235	-22.7885	0.1088	0.8482	4.8559
3	-0.0223	-0.0003	0.0005	-0.0266	-76.6204	-0.0150	0.3701	3.7974
4	0.4144	-0.0018	-0.0046	0.0357	-222.9720	1.2176	2.6398	23.4144
5	0.0155	-0.0002	-0.0001	0.0084	18.4710	-0.0707	0.0359	-3.1046
6	-0.3306	-0.0007	0.0028	-0.0422	0.0246	0.6163	-1.6640	3.6929
7	12.458	-0.0025	-0.0034	-0.0576	-0.0499	1.0368	1.5311	5.4053
8	0.0000	-0.0002	0.0000	0.0026	0.0000	-0.0137	0.0000	0.2817
9	0.0365	-0.0008	0.0030	0.0495	4.7194	0.2357	-1.0777	-17.7164
10	0.0338	0.0004	-0.0003	0.0203	-26.0544	-0.4205	1.4586	-21.1434
11	0.2010	-0.0001	-0.0006	0.0008	-64.2467	-0.1078	0.1749	5.5097
12	0.1183	0.0001	-0.0002	0.0187	-60.3032	0.0186	0.1041	18.9206
13	0.0229	-0.0006	0.0003	0.0102	73.5939	-0.0914	-0.3023	-2.9522
14	0.0867	-0.0006	-0.0005	0.0221	-3.6213	0.3328	0.5946	-14.1480
15	0.1036	-0.0002	-0.0001	0.0083	-26.1434	0.0260	0.0528	2.2723
16	0.1267	0.0005	-0.0001	0.0067	-97.0739	0.0317	0.2886	7.2859

표 1는 생성된 퍼지규칙의 출력값을 결정짓는 파라미터이다.

4. 시뮬레이션

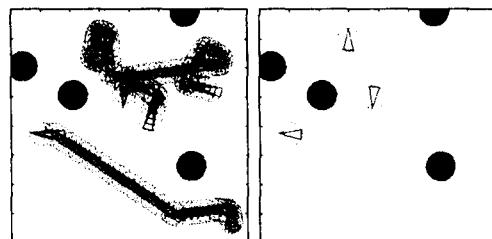
장애물을 회피하는 자율이동로봇을 구현하여 알고리즘을 이용한 시뮬레이션과 퍼지규칙을 이용한 시뮬레이션을 비교하고 성능을 비교하였다.



(a) 알고리즘 (b) 퍼지 시스템

그림 5: 로봇의 장애물 회피 알고리즘

그림 5(a)는 자율이동로봇의 장애물 회피 알고리즘을 나타낸 것으로써 자율이동로봇이 랜덤하게 주어진 위치에서 앞으로 전진이라는 행동패턴을 가지고 있고 자신의 행동패턴으로 전진을 실행하다가 장애물이 존재하면 회피하여 장애물로 둘러싸인 부분에서 탈출하는 것을 나타내었다. 그림 5(b)는 같은 조건의 자율이동로봇을 클러스터링으로 생성된 퍼지규칙을 이용하여 시뮬레이션 한 결과로 같은 행동패턴을 구현할 수 있음을 알 수 있다. 그림 6은 장애물 회피 알고리즘을 이용하여 다수의 자율이동로봇이 존재할 때 개체가 다른 자율이동로봇이나 장애물과 근접하였을 때 장애물로 인식하고 회피하는 것을 나타내었다.



(a) 이동 궤적 (b) 최종 결과

그림 6: 알고리즘을 이용한 3개의 로봇

그림 6(a)는 알고리즘에 이용한 자율이동로봇의 이동 궤적을 나타낸다. 그림 6(b)는 그림 6(a)의 마지막 최종 결과를 나타내고 있다.

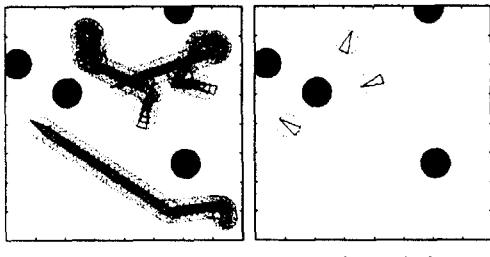


그림 7: 퍼지 시스템을 이용한 3개의 로봇

그림 7(a)은 같은 조건의 자율이동로봇을 클러스터링하여 생성된 퍼지규칙을 이용하여 시뮬레이션 한 자율이동로봇의 이동 궤적을 나타내며 그림 7(b)는 최종 결과를 나타낸다. 시뮬레이션으로 알고리즘과 같은 움직임을 구현 할 수 있음을 알 수 있다.

그림 8는 두 번째 시뮬레이션의 장애물 회피 알고리즘을 이용한 자율이동로봇과 퍼지규칙을 이용한 자율이동로봇의 성능 비교를 나타내었다.

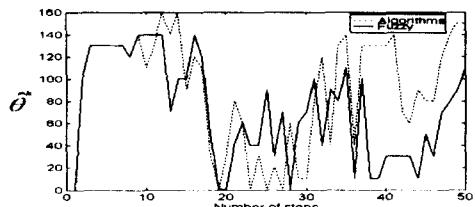


그림 8: 자율이동로봇의 성능비교

그림 8는 매 단계마다 자율이동로봇들의 센서 입력의 평균값으로 나타내었다. 퍼지규칙에 의한 자율이동로봇이 알고리즘에 의한 자율이동로봇보다 단순한 움직임을 보이고 있음을 알 수 있다.

그림 6 과 그림 7의 결과를 통해 클러스터링 된 퍼지규칙으로 장애물 회피 알고리즘과 같은 자율이동로봇을 설계할 수 있음을 알 수 있다. 그림 8의 결과를 통해 자율이동로봇의 성능이 향상되었음을 알 수 있다.

5. 결 론

12개의 센서로 상대적인 거리와 각도만을 이용하여 장애물을 회피하는 알고리즘을 갖는 자율이동로봇으로부터 입출력 데이터를 수집하고 클러스터링하여 퍼지규칙을 생성함으로써 비슷한 행동 특성을 보여주는 자율이동로봇을 설계하였고 성능에 대한 두 자율이동로봇을 비교하였다. 앞으로 연구되어야 할 방향은 센서에 의한 자율이동로봇의 군집 및 정렬을 규칙으로 표현함으로써 보다 군집능에 행동 특성에 가까운 자율이동로봇을 설계하는 것이다.

참고 문헌

- [1] C. G. Langton, (ed.) *Artificial Life II*, Addison Wesley, 1992.
- [2] M. Sipper, "An Introduction to Artificial Life," *Explorations in Artificial Life*, pp. 4-8, 1995.
- [3] C. W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model," *Computer Graphics*, Vol. 21, No. 4, pp. 25-34, July 1987.
- [4] D. Terzopoulos, X. Tu, and R. Grzeszczuk, "Artificial Fishes: Autonomous Locomotion, Perception, Behavior, and Learning in a Simulated Physical World," *Artificial Life I*, pp. 327-351, 1994.
- [5] J. Xiao, Z. Michalewicz, L. Zhang, and K. Trojanowski, "Adaptive Evolutionary Planner/Navigator for Mobile Robots", *IEEE Trans. Evolutionary Computation*, Vol. 1, No. 1, pp. 18-28, April 1997
- [6] Moshe Sipper, "An Introduction To Artificial Life", *Explorations in Artificial Life(special issue of AI Expert)*, pp. 4-8, September 1995.
- [7] 김장현, 공성곤, "자율이동로봇의 정렬 군집능 알고리즘 구현", 대한 전기학회

하계학술대회 논문집, G권, pp. 2186-2188,
7. 1998

- [8] 김장현, 공성곤, “자율이동로봇의 정렬
군지능 알고리즘 구현”, 대한 전기학회
하계학술대회 논문집, G권, pp.2186-2188,
7. 1998