

퍼지 분류자 시스템을 이용한 자율이동로봇의 충돌 회피 학습

Learning Rules for AMR of Collision Avoidance using Fuzzy Classifier System

*반창봉, 전효병, 심귀보
중앙대학교 전자전기공학부

Tel : 02-820-5319; Fax : 02-817-0553; E-mail: kbsim@cau.ac.kr

*Chang-Bong Ban, Hyo-Byung Jun, Kwee-Bo Sim
School of Electrical and Electronics Engineering, Chung-Ang University
Tel : 02-820-5319; Fax : 02-817-0553; E-mail: kbsim@cau.ac.kr

A Classifier System processes a discrete coded information from the environment. When the system codes the information to discontinuous data, it loses excessively the information of the environment. The Fuzzy Classifier System(FCS) makes the classifier system be able to carry out the mapping from continuous inputs to outputs. It is the FCS that applies this ability of the machine learning to the concept of fuzzy controller. It is that the antecedent and consequent of classifier is same as a fuzzy rule of the rule base. In this paper, the FCS is the Michigan style and fuzzifies the input values to create the messages. The system stores those messages in the message list and uses the implicit Bucket Brigade Algorithms. Also the FCS employs the Genetic Algorithms (GAs) to make new rules and modify rules when performance of the system needs to be improved. We will verify the effectiveness of the proposed FCS by applying it to AMR avoiding the obstacle.

I. 서 론

분류자 시스템은 Holland에 의해서 제안된 기계학습법(machine learning)이다. 분류자 시스템은 문장적 의미를 가지는 간단한 문자열 규칙을 학습과 진화 기법을 이용하여 실시간으로 학습하는 방법이다[1]. 또한, 외부환경으로부터 강화 신호에 의해 학습이 이루어지기 때문에 강화학습(reinforcement learning)의 한 부류가 된다. 분류자 시스템은 인간의 사고 능력을 모델링하는 실시간 학습 방법으로 주목을 받고 있지만, 이산적인 정보만을 다루어 정보의 손실이 문제점으로 인식되고 있다[2]. 이 점을 보완하기 위해 퍼지 분류자 시스템이 제안되었다[3]. 그러나 기존의 퍼지 분류자 시스템은 교사 학습의 성격이 강하다. 일반적으로 퍼지 시스템에서의 출력은 하나의 규칙에 의한 것이라기 보다는 다수의 규칙에 의해서 출력이 정해지는 것이 더 적합하다.

본 논문에서는 유용한 규칙의 집합을 학습하는 퍼지 분류자 시스템을 제시하였다. 제안된 시스템을 자율이동로봇(Autonomous Mobile Robot)의 장애물 회피학습에 적용하여 그 유용성을 확인하였다.

II. 퍼지 분류자 시스템

분류자 시스템은 외부 환경에 대한 정보를 이산적인 코드로 변환하여 취급한다. 이 경우 실제 외부 환경에 대한 정보를 상실할 가능성이 높다. 이 점을 보완하기 위해 퍼지 분류자 시스템은 외부 환경에 대한 정보를 퍼지화하여 취급하고, 이 값을 퍼지 분류자를 통해서 연속적인 출력 값으로 매핑한다. 즉, 분류자 시스템에 퍼지 제어기의 개념을 접목한 것이 퍼지 분류자 시스템이다.

본 논문에서는 미시간(Michigan)방법의 퍼지 분류자 시스템[4]을 설명한다. 외부 환경에 대

한 정보를 퍼지화하여 입력 메시지를 생성하고 메시지 리스트를 만들고 퍼지 분류자 리스트 내의 퍼지 분류자들과의 정합과정을 통해서 출력값을 얻어낸다. 내재적 Bucket Brigade 알고리즘을 사용하여 퍼지 분류자들간의 관계를 정립한다. 또한, 일정 시간동안 시스템의 향상이 없으면 새로운 퍼지 분류자 생성을 시도한다. 그림 1은 본 논문에서 제안된 퍼지 분류자 시스템의 구조도를 보여준다.

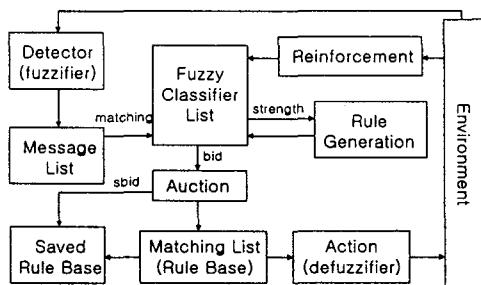


그림 1. 퍼지 분류자 시스템의 구조

1. 메시지와 메시지 리스트

Detector는 퍼지 논리 제어기의 퍼지화기와 같이 외부 환경에 대한 정보를 퍼지화하여 메시지를 생성한다. 생성된 메시지는 메시지 리스트에 저장된다. 그림 2는 퍼지화된 메시지의 생성과 형식을 보여주고 있다.

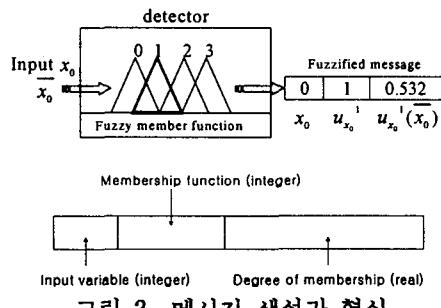


그림 2. 메시지 생성과 형식

메시지는 3부분으로 구성되어 있다. 처음 부분은 입력변수를, 나머지 두 부분은 입력변수가 소속된 소속함수와 소속정도를 나타낸다. 예를 들어 입력 값 \bar{x}_0 가 첫 번째와 두 번째 소속함수에 소속정도 0.6 ($= u_{x_0}^0(\bar{x}_0)$), 0.4 ($= u_{x_0}^1(\bar{x}_0)$)인 값을 각각 가질 때 입력 값 \bar{x}_0 이 detector를 통과했을 때 생성되는 메시지는 0:0:0.6, 0:1:0.4가 된다. 메시지의 마지막 부분의 소속정도는 정합과정에서 분류자의 발화 강도(firing strength)를 결정할 때 사용된다.

2. 퍼지 분류자 리스트와 퍼지 분류자

퍼지 분류자 리스트는 퍼지 분류자들을 저장하는

공간이다. 퍼지 분류자는 IF-THEN규칙과 신뢰도 값으로 표현된다. 퍼지 분류자의 조건부에는 각 입력 변수의 소속함수를 표시하고, 행동부에는 각 출력 변수의 소속함수를 표시한다. 일반적인 규칙을 표현하기 위해 입력변수에 대한 소속함수를 "don't care"로 표시 할 수도 있다. 입력 변수가 k 개 존재하면 퍼지 분류자의 조건부에는 k 개의 조건이 존재한다. 신뢰도 값은 각 퍼지 분류자의 유용성을 나타낸다.

3. 메시지 정합

시스템은 메시지 리스트에 저장된 메시지를 만족하는 퍼지 분류자를 찾는다. 이 과정을 메시지 정합이라고 한다. k 개의 입력 변수가 존재하면 퍼지 분류자의 조건부에 k 개 조건 존재한다. 또한, 각 입력변수에 대해 하나 이상의 메시지가 존재한다. 각 메시지의 입력 변수 값은 퍼지 분류자 조건부의 어떤 조건과 비교를 해야하는지 알려준다. 분류자의 k 개의 조건이 메시지 리스트에 저장된 임의의 k 개의 메시지와 일치하면 퍼지 분류자가 정합 된 것이다. 이 때 메시지의 소속정도 값 중 가장 작은 값을 정합된 퍼지 분류자의 발화강도로 정한다. 그림 3은 메시지 정합과정을 보여준다. 그림 3에서 분류자를 정합하는 4개의 메시지가 각각 $u_{x_0}(\bar{x}_0), u_{x_1}(\bar{x}_1), u_{x_2}(\bar{x}_2), u_{x_3}(\bar{x}_3)$ 의 소속정도를 가지고 있다면 발화 강도는 식 (1)과 같다.

$$F_K = u_{x_0}(\bar{x}_0) \wedge u_{x_1}(\bar{x}_1) \wedge u_{x_2}(\bar{x}_2) \wedge u_{x_3}(\bar{x}_3) \quad (1)$$

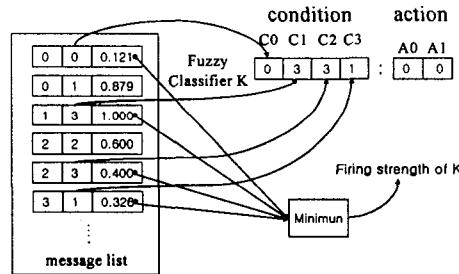


그림 3 메시지 리스트와 퍼지 분류자의 정합

4. 내재적 Bucket Brigade 알고리즘

정합된 퍼지 분류자들은 Rule Base에 참여하기 위해 bid값을 제시한다. bid값은 식 (2)와 같이 퍼지 분류자의 신뢰도 값(S_r)과 bid 상수(C_{bid}), 발화강도(F_r)의 곱으로 표시된다.

$$Bid_r = C_{bid} \cdot S_r \cdot F_r \quad (2)$$

제시된 bid값에 확률적으로 비례해서 N 개의 규칙을 선택하여 Rule Base를 구성한다. Rule Base를 구성한 퍼지 분류자들은 제시한 bid값을 신뢰도에서 제거한다. Rule Base를 구성하는 퍼지 분류자는 퍼지 제어기에서 결과 값을

얻는 것과 같은 방법으로 시스템의 출력을 얻어낸다. t 시간에 구성된 Rule Base의 퍼지 분류자들은 간접적으로 $t+1$ 시간에 구성되는 Rule Base에 영향을 준다. 이것을 내재적 Bucket Brigade 알고리즘이라고 한다. 그림 4는 내재적 Bucket Brigade 알고리즘의 구조도를 보여 주고 있다.

내재적 Bucket Brigade 알고리즘을 시스템에 적용하기 위해서식 (3)과 같이 t 시간의 Rule Base를 구성한 모든 분류자들이 제시한 bid값을 더한다.

$$S_{bid}(t) = \sum_{k \in M(t)} Bid_k \quad (3)$$

이 값을 bid합 이전의 Rule Base를 구성한 분류자들에게 발화 강도에 비례하여 분배한다. 이와 같은 Bucket Brigade 알고리즘은 식 (4)와 같이 표현된다.

$$S_r(t) = S_r(t-1) - Bid_r + \frac{F_r}{\sum_{i \in M(t)} F_i} S_{bid}(t) \quad (4)$$

여기서 $M(t)$ 는 $t-1$ 시간에 Rule Base를 구성한 퍼지 분류자의 인덱스 집합을 나타낸다.

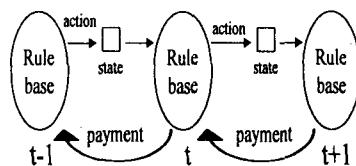


그림 4 내재적 Bucket Brigade 알고리즘

5. 자연 강화 학습

t 시간의 출력이 시스템 전체에 큰 영향을 주지 못하는 경우가 있다. 이 경우 시간의 간격을 주어 충분한 출력이 이루어진 후 강화 학습을 실행하게 된다[5]. 즉, 시스템은 T 시간이 지난 후에 강화신호를 받아서 T 시간 동안 출력에 영향을 준 모든 퍼지 분류자에게 발화 강도에 비례하여 강화신호를 분배한다. 강화신호의 분배는 식 (5)와 같이 한다.

$$S_r(t) = S_r(t-1) + \sum_{p \in L(t), q \in B(t)} \frac{F_r(p)}{\sum_{p \in L(t), q \in B(t)} F_q(p)} R(t) \quad (5)$$

여기서 $L(t)$ 와 $B(t)$ 는 각각 t 스텝에서 과거 T 시간 동안 입력 상태집합과 Rule-Base에 참여한 규칙 집합을 나타내고 $F_r(p)$ 는 규칙 r 이 p 상태일 때 규칙의 발화 강도를 나타낸다.

Rule Base에 참여하지 않는 퍼지 분류자들을 제거하기 위해 강화신호를 받은 시점에서 퍼지 분류자 리스트에 존재하는 모든 분류자들에게 식 (6)과 같은 tax를 부과한다. C_{tax} 는 tax 상수 값이며,

$C_{tax} \ll C_{bid}$ 을 만족한다.

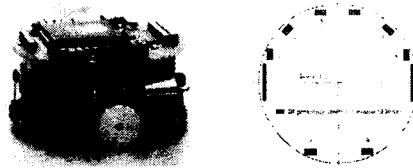
$$S_r(t) = S_r(t-1) - C_{tax} S_r(t-1) \quad (6)$$

6. 퍼지 분류자의 생성

일정 시간이 지나도 시스템 성능의 향상이 없으면 새로운 퍼지 분류자의 생성을 시도한다. 퍼지 분류자 시스템은 가장 적합한 하나의 규칙을 찾는 것이 아니라 적합한 규칙의 집합을 찾는 것을 목적으로 한다. 이를 위해 crowding replacement 방법[1]을 사용한다.

III. 실험

본 논문에 제시된 퍼지 분류자 시스템을 자율이동로봇인 Khepera(그림 5(a))에 직접 적용하여 실험하였다.



(a) Khepera 로봇 (b) 로봇의 센서 구성
그림 5. 자율이동로봇의 외형과 센서 구성

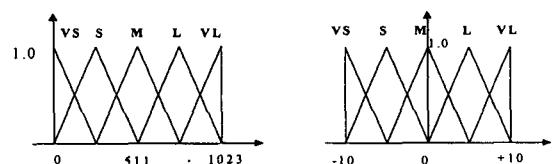
자율이동로봇은 그림 5(b)와 같이 $\pm 30^\circ$ 범위를 감지할 수 있는 8개의 센서를 통해서 외부 환경에 대한 정보를 얻는다. 실험에서는 0~5번의 센서를 2개씩 그룹을 지어 사용하였다. 예를 들어 센서 0과 1을 S0라는 하나의 그룹으로 외부 환경에 대한 정보를 얻는다. 출력은 자율이동로봇의 두 모터를 구동하게 된다. 표 1은 입출력 변수의 범위를 나타낸다. 센서 값이 큰 경우 장애물이 가까이 있음을 나타낸다. 만약 센서 값이 1023이면 자율이동로봇이 장애물과 충돌한 것을 의미한다. 모터 값은 (+)는 전진을 (-)는 후진을 의미한다.

표 1. 입출력 변수값의 범위

입력 변수			출력 변수	
S0(0,1)	S1(2,3)	S2(4,5)	LM	RM
0~1023	0~1023	0~1023	-10~+10	-10~+10

단위 값 당 센서의 감지 거리: 약 0.05mm

단위 값 당 모터의 속도: 0.08mm/10ms



1)입력 변수의 소속 함수

2)출력 변수의 소속함수

VS:Very Small, S:Small, M:Medium, L:Large, VL:Very Large

그림 6. 입출력 변수의 소속 함수

자율이동로봇은 센서를 통해 얻어진 정보를 이용

하여 장애물을 회피하며 이동할 수 있는 적절한 퍼지 규칙을 학습한다. 그림 6은 입력 변수와 출력 변수에 대한 퍼지 집합이다. 각각 5개의 소속함수로 구성하였다. 퍼지 분류자의 조건부에는 don't care 문자를 포함 6개의 소속함수 값 중 하나를 택할 수 있으므로 216가지의 경우의 수가 있고, 행동부에는 5개의 소속함수 값 중 하나를 택할 수 있으므로 25 가지의 경우의 수가 있다. 즉, 이 시스템의 퍼지 분류자의 탐색 영역은 5400개의 퍼지 분류자 후보를 갖는다. 퍼지 분류자 시스템은 이 탐색 영역 후보 중에서 자율이동로봇이 장애물과 충돌하지 않고 이동하기 위한 적절한 규칙의 집합을 찾게 된다.

자율이동로봇이 센서로부터 정보를 얻고 퍼지 분류자 시스템을 통해 얻은 출력으로 한번 이동하는 것을 1 스텝이라고 한다. 이 실험에서는 5 스텝후 강화 신호를 받고 10번의 강화 신호를 받은 후 rule discovery를 수행한다. 150스텝을 1회 실행으로 한다. 1회 실행이 끝나면 원래 위치로 되돌아가서 다시 학습을 수행한다. 강화 신호는 식 (7)과 같다.

$$R(t) = \alpha D(t) - \beta CO(t) \quad (7)$$

$D(t)$ 는 5 스텝동안 이동한 거리를, $CO(t)$ 는 충돌 횟수를 나타낸다. 실험의 주요 시뮬레이션 파라미터 값은 초기 분류자의 개수 : 100 개, Maximum Step : 150 Step, 실행 횟수 : 1000 회, 분류자의 초기 신뢰도 : 100, Bid 상수 : 0.1, Tax 상수 : 0.01, 교차 확률 : 0.8, 돌연변이 확률 : 0.01로 각각 설정하였다.

실험 결과 300회 실행후 자율이동로봇이 장애물에 충돌하지 않고 이동하였다. 각 실험마다 얻어진 퍼지 분류자들은 조금씩 차이를 보였지만 자율이동로봇은 유사한 행동을 보여 주었다. 퍼지 분류자 리스트에는 많은 퍼지 분류자들이 존재하지만 이중에서 20~25개만이 충돌하지 않고 목표지점에 도착하는 행동에 사용되었다. 실험결과 중 한 가지를 표 2의 유용한 퍼지 분류자의 집합에 나타내었다.

표 2. 유용한 퍼지 분류자 집합

	조건부			행동부	
	S0	S1	S2	LM	RM
1	L	L	VS	VL	S
2	S	L	VS	M	S
3	VS	S	VS	S	M
4	S	S	VS	VL	VL
5	S	S	VS	VL	S
6	L	VS	VS	VL	M
7	VS	VS	S	M	VS
8	VS	S	S	VL	M
9	M	M	M	S	S
10	M	VS	VS	M	S
11	S	M	VS	M	M
12	L	VS	VS	VL	L
13	L	VL	M	VS	VL
14	VL	L	VS	VS	S
15	S	S	M	M	S

16	VS	L	VS	VL	M
17	M	M	VS	VL	M
18	VL	VS	S	VL	L
19	VS	VS	M	L	VL
20	VL	M	VS	L	VS

IV. 결론

학습과 진화의 기법을 접목하여 인간의 사고 능력을 모델링한 분류자 시스템은 실시간 학습 방법으로 주목을 받고 있다. 하지만, 이 산적인 정보를 다룸으로서 정보 손실이 중요한 문제로 생각되어졌다. 그러나, 퍼지 분류자 시스템은 분류자 시스템에 퍼지 제어기의 개념을 도입하여, 연속적인 입력 값을 연속적인 출력 값으로 매핑할 수 있도록 해 준다. 본 논문에서 제안된 퍼지 분류자 시스템은 입력 값을 퍼지화 하여 메시지를 생성하고, Bucket Brigade 알고리즘과 자연 강화학습을 통해 유용한 퍼지 분류자들을 찾아낸다. 또한, 시스템의 성능 향상을 위해 새로운 퍼지 분류자의 생성을 시도하기도 한다. 제안된 퍼지 분류자 시스템을 자율이동로봇의 장애물 회피 학습에 적용하여 그 유용성을 확인하였다.

감사의 글

본 연구는 과학기술부의 뇌과학 프로젝트 (Braintech21)의 지원으로 이루어진 결과임. (과제 번호 : 98-J04-01-01-A-07)

V. 참고 문헌

- [1] Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.
- [2] Whitehead, S. D., & Ballard, D.H., "Learning to perceive and act by trial and error," *Machine Learning*, 7, pp. 45-83
- [3] Valenzuela-Rendon, M., "The Fuzzy classifier system: Motivations and first results," *Parallel Problem Solving from Nature - PPSNII*, Springer-Verlag, pp. 330-334, 1991.
- [4] Dorigo, M., and Bersini, H., "A Comparison of Q-Learning and Classifier Systems," Proc. of From Animals to Animats, Third International Conference on Simulation of Adaptive Behavior, 1994.
- [5] Bonarini, A., "Anytime learning and adaptation of hierarchical fuzzy logic behaviors," *Adaptive Behavior Journal*, Special Issue on Complete Agent Learning in Complex Environments, Vol 5(3-4), pp. 281-315, 1997.