

# 퍼지 분류자 시스템을 이용한 자율이동로봇의 충돌 회피학습

## Learning Rules for AMR of Collision Avoidance using Fuzzy Classifier System

반창봉 · 심귀보

Chang-Bong Bam and Kwee-Bo Sim

중앙대학교 전자전기공학부

### 요 약

본 논문에서는 연속적인 입력을 연속적인 출력으로 매핑하는 것을 가능하게 하는 퍼지 분류자 시스템을 제안한다. 퍼지 분류자 시스템은 기계학습의 방법중 하나인 분류자 시스템을 퍼지 제어기의 개념에 적용한 것이다. 즉 분류자의 조건부는 퍼지 규칙의 전건부와 행동부는 후건부와 같은 행태가 된다. 퍼지 분류자 시스템은 입력 값을 퍼지화된 메시지로 변환하고 메시지 리스트에 저장한다. 저장된 메시지와 퍼지 분류자 리스트의 분류자들과 정합과정을 통해 룰-베이스를 구성하고, 퍼지 분류자들의 유용성을 검증하기 위해 버킷 릴레이 알고리즘을 적용한다. 또한 새로운 규칙을 생성하거나 규칙을 수정하여 시스템의 성능을 향상시키기 위해 유전 알고리즘을 사용한다. 이러한 과정을 통해 유용한 규칙집합을 찾아내고, 시스템은 그 규칙들에 의해 출력 값을 내보낸다. 제안된 퍼지 분류자 시스템을 자율이동로봇의 충돌 회피 학습에 적용하여 그 유용성을 확인하였다.

### ABSTRACT

In this paper, we propose a Fuzzy Classifier System(FCS) makes the classifier system be able to carry out the mapping from continuous inputs to outputs. The FCS is based on the fuzzy controller system combined with machine learning. Therefore the antecedent and consequent of a classifier in FCS are the same as those of a fuzzy rule. In this paper, the FCS modifies input message to fuzzified message and stores those in the message list. The FCS constructs rule-base through matching between messages of message list and classifiers of fuzzy classifier list. The FCS verifies the effectiveness of classifiers using Bucket Brigade algorithm. Also the FCS employs the Genetic Algorithms to generate new rules and modify rules when performance of the system needs to be improved. Then the FCS finds the set of the effective rules. We will verify the effectiveness of the proposed FCS by applying it to Autonomous Mobile Robot avoiding the obstacle and reaching the goal.

### 1. 서 론

분류자 시스템은 주어진 작업을 완수하기 위해 환경과의 상호작용을 통해서 학습을 하는 적응 시스템이다. 또한 환경으로부터 받는 강화 신호에 의해서 학습이 이루어지므로 강화학습(reinforcement learning)시스템의 한 부류가 된다. 분류자 시스템은 문장적 의미를 가지는 간단한 문자열 규칙을 학습시키는 기계학습법(machine learning)[1]으로, Holland에 의해서 제안되었다. Holland는 Bucket Brigade 알고리즘을 사용하여 각 분류자의 유용성을 학습하도록 하고, 유전 알고리즘을 사용하여 새로운 규칙을 생성하거나 유용하지 못한 규칙을 제거하는 것이 가능하도록 하였다[2]. 그러나 Holland의 분류자 시스템은 환경으로부터의 정보를 비트로 표현되는 이산적인 코드의 학습을 수행한다. 실제로 대부분의 시스템은 연속적인 데이터를 취급한다. 그런데 연속적인 데이터를 이산적인 코

드로 취급하면 대부분의 정보를 상실하게 된다[3].

퍼지 분류자 시스템은 이러한 문제에 대해서 입력 값을 퍼지 집합으로 분할하고 퍼지 제어기의 개념을 도입하여 연속적인 입력에서 연속적인 출력으로 매핑을 실현하는 시스템이다. 즉, 분류자의 조건부와 행동부가 퍼지 집합으로 확장된 것이다. 퍼지 분류자 시스템은 Valenzuela-Rendor에 의해서 처음으로 제안되었다[4]. 그의 시스템은 Michigan 방법을 기반으로 하여 개개의 규칙에 신뢰도를 할당하는 방법을 사용하고 1 입력 1출력 함수 근사 문제에 적용하였다. 앞의 연구가 미리 정한 소속함수를 이용한 반면 Parodi와 Bonelli는 실수를 사용하여 소속함수의 모양도 같이 학습하는 방법을 사용하였다. 또한 메시지 리스트와 Bucket Brigade 알고리즘을 사용하지 않고, 신뢰 할당 절차를 간략화하고 있다[5]. 하지만 위의 연구들은 각각의 입력에 대한 정답이 주어지고 있는 교사 학습이기 때문에 강화학습 방법의 적용으로 보기에는 힘들

다. 일반적으로 퍼지 시스템에서의 출력은 하나의 규칙에 의한 것이라기 보다는 다수의 규칙에 의해서 출력이 정해지는 것이 더 적합하다. Furuhashi는 다중 stimulus-response Michigan 방법의 퍼지 분류자 시스템을 사용하여 배의 조선편제에 적용하였다[6]. 퍼지 분류자 시스템에서는 퍼지 규칙에 대해서 기계학습이 이루어지기 때문에 퍼지 규칙의 코딩, 메시지의 퍼지화, 퍼지규칙을 이용한 행동결정, 퍼지 규칙에 대한 신뢰도 할당, 유전자 알고리즘에 의한 규칙의 갱신 등에 대해서 고려해야한다.

본 논문에서는 분류자 시스템에서의 메시지 리스트를 사용하고 내재적인 Bucket Brigade 알고리즘을 적용하였다. 실수의 입력 값을 detector를 통과시켜 입력 변수가 포함되는 소속함수와 그 소속정도를 나타내는 퍼지화 된 메시지를 생성한다. 하나의 입력 변수에 대해서 퍼지화 된 메시지는 그 소속함수의 모양에 따라 하나 이상의 메시지를 생성할 수 있다. 생성된 메시지는 메시지 리스트에 저장된다. 저장된 메시지를 토대로 메시지를 만족하는 규칙을 퍼지 분류자 리스트에서 찾는다. 만족하는 규칙들은 퍼지 rule-base에 참가하기 위해 각 규칙이 보유하고 있는 신뢰도를 토대로 서로 경합을 한다. 마지막으로 rule-base에서 제어 행동을 만들어낸다. 일정한 스텝동안 제어행동을 수행한 후에 환경으로부터 강화신호를 받아서 제어행동에 관련된 규칙들에게 분배한다[7]. 이어 시스템의 성능의 향상이 없으면 새로운 규칙을 생성하기 위해서 유전연산을 수행한다. 여러 개의 규칙집합 후보 중에서 가장 좋은 규칙 집합을 찾는 것이 아니라 하나의 규칙집합을 이용하여 유용한 규칙집합을 얻는 것이 가능하다. 본 논문에 제안된 방법을 자율이동로봇 (Autonomous Mobile Robot)에 적용하여 그 유용성을 확인한다.

## 2. 퍼지분류자 시스템

Michigan 방법에 의한 Genetic Based Machine Learning(GBML)을 분류자 시스템(classifier system)이라고 부르고, 1976년에 Holland[8]에 의해서 처음 제안되었다. GBML은 규칙의 표현방법에 따라 Michigan 방법과 Pittsburgh 방법으로 구분한다. Pittsburgh 방법은 완전한 규칙의 집합을 하나의 스트링으로 표현하며 Michigan 방법은 하나의 규칙을 하나의 스트링으로 표현하여 진화한다. 즉 Michigan 방법에서는 규칙집합을 하나만 가지고 Pittsburgh 방법은 여러 개의 규칙집합을 보유한다.

분류자 시스템은 환경으로부터의 정보를 비트로 표

현되는 이산 적인 코드로 학습을 수행한다. 연속적인 데이터를 이산 적인 코드로 취급하면 대부분의 정보를 상실할 수 있다. 이 점을 보완하기 위하여 퍼지 분류자 시스템은 분류자 시스템을 이용하여 연속적인 입력 값을 연속적인 출력 값으로 매핑해주는 기능을 가지고 있다. 분류자 시스템은 분류자로 표현되는 규칙의 유용성을 반복되는 학습을 통하여 알아낼 수 있는 장점을 가지고 있다. 이러한 기계 학습적 기법을 퍼지 제어기의 개념에 접목한 것이 퍼지 분류자 시스템이라고 할수 있다.

본 논문에서는 Michigan 방법의 퍼지 분류자 시스템을 설명한다. 입력 값을 퍼지화를 수행하여 메시지를 생성하여 메시지 리스트를 만들고 내재적 Bucket Brigade 알고리즘을 사용한다. 그림 1은 본 논문에서 제안된 퍼지 분류자 시스템의 구성도를 보여준다.

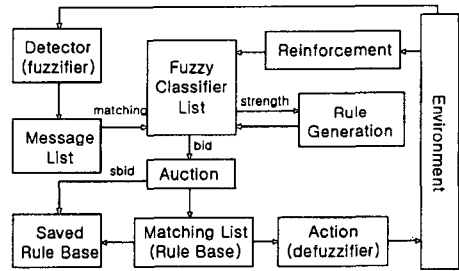


그림 1. 퍼지분류자 시스템의 구조  
Fig. 1. Structure of Fuzzy Classifier System

### 2.1 메시지와 메시지 리스트

검지기(detector)는 환경으로부터 각 입력변수 값을 받아서 퍼지화를 수행한다. 검지기는 퍼지 논리 제어기의 퍼지화기(fuzzifier)와 같은 역할을 한다. 퍼지화된 메시지는 메시지 리스트에 저장된다. 그림 2는 퍼

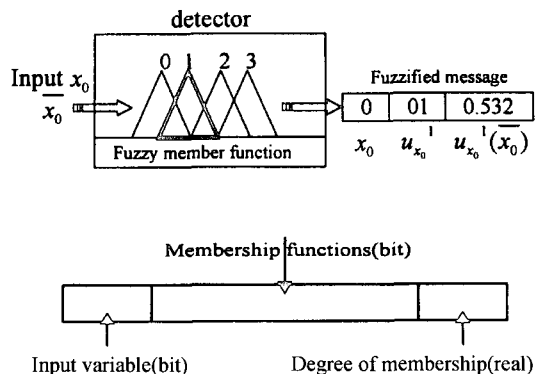


그림 2. 메시지 형식과 생성  
Fig. 2. Construction and format of a message

지화된 메시지의 형식을 나타낸다.

메시지는 3부분으로 이루어진다. 처음 부분은 입력 변수를 표시하고 두 번째는 그 입력변수의 실수 값에 소속정도를 가지는 소속함수를 나타낸다. 마지막 부분은 그때의 소속정도를 0과 1사이의 실수 값으로 표시한다. 예를 들어 입력 벡터  $x$ 에 대해서 다음과 같이 정의한다.

$$x = \left\{ \left( x_i, U_i \left\{ T_{x_i}^0, T_{x_i}^1, \dots, T_{x_i}^{k_i} \right\}, \left\{ u_{x_i}^0, u_{x_i}^1, \dots, u_{x_i}^{k_i} \right\} \right) \right\}_{i=0, \dots, n} \quad (1)$$

여기서  $x_i$ 는 입력변수,  $U_i$ 는 universe of discourse,  $T(x_i) = \{T_{x_i}^0, T_{x_i}^1, \dots, T_{x_i}^{k_i}\}$ 는 언어항 집합,  $u(x_i) = \{u_{x_i}^0, u_{x_i}^1, \dots, u_{x_i}^{k_i}\}$ 는 소속함수를 나타낸다. 입력  $x_0$ 가 있고 각각 4개의 언어항과 소속함수가 정의되어 있다고 가정한다. 입력  $x_0$ 의 crisp 입력값  $\bar{x}_0$ 이 첫 번째와 두 번째 소속함수에 각각  $u_{x_0}^0(\bar{x}_0) = 0.6$ ,  $u_{x_0}^1(\bar{x}_0) = 0.4$ 의 소속 정도를 가지고 있을 때 crisp 입력 값  $\bar{x}_0$ 가 detector를 통과했을 때 생성되는 메시지는 그림 2와 같은 형식으로 00:00:0.6, 00:01:0.4가 된다. 각 메시지가 소유하고 있는 실수 값은 매칭 과정에서 분류자의 발화 강도(firing strength)를 결정할 때 사용된다.

### 2.2 분류자 리스트

분류자 리스트의 분류자는 조건부에 각 입력변수의 소속함수와 행동부에 각 출력변수의 소속함수를 bit string으로 표시하여 IF-THEN 규칙을 생성한다. 퍼지 분류자 시스템의 입력변수가  $k$ 개 존재한다면 분류자의 조건부에는  $k$ 개의 조건이 존재한다. 각 분류자는 유용성을 가를 수 있게 하는 신뢰도 값을 소유하고 학습을 통해서 그 값을 변화시킨다.

### 2.3 내재적 Bucket Brigade 알고리즘

검지기가 입력변수 값을 메시지로 변환하고 그것을 메시지 리스트에 저장하면, 시스템은 메시지 리스트에 있는 메시지를 만족하는 분류자를 찾는다. 이 과정을 정합(Matching)과정이라고 부른다. 입력 변수가  $k$ 개 존재하면 메시지 리스트에는 각 입력 변수에 대해서 하나이상의 메시지가 저장되어있고, 분류자 리스트의 분류자는  $k$ 개의 조건을 가진다. 메시지의 각 입력변수 비트는 분류자의 몇 번째 조건과 비교를 해야 하는지를 알려준다. 분류자의 각 조건과 메시지의 소속함수 비트와 비교하여 모두 일치하면 분류자가 정

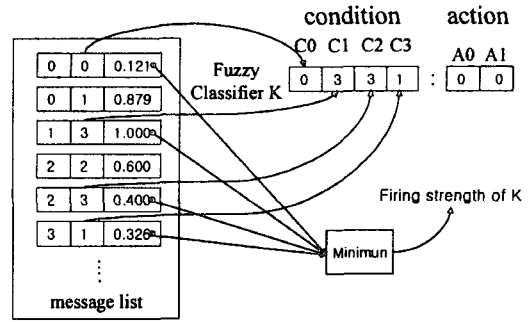


그림 3. 메시지 리스트와 분류자의 Matching  
Fig. 3. Message list and Matching classifier

합 된 것이다. 이때 퍼지 제어기의 추론 부에서 발화 강도를 구하는 것과 같은 방법으로 각 메시지의 소속 정도 값의 최소 값을 정합된 분류자의 발화 강도로 정한다. 그림 3에서 분류자와 정합하는 4개의 메시지가 각각  $u_{x_0}(\bar{x}_0)$ ,  $u_{x_1}(\bar{x}_1)$ ,  $u_{x_2}(\bar{x}_2)$ ,  $u_{x_3}(\bar{x}_3)$  소속정도를 가지고 있다면 발화 강도는 다음 식과 같다.

$$F_K = u_{x_0}(\bar{x}_0) \wedge u_{x_1}(\bar{x}_1) \wedge u_{x_2}(\bar{x}_2) \wedge u_{x_3}(\bar{x}_3) \quad (2)$$

정합된 분류자들은 rule-base에 참여하기 위해서 분류자 시스템과 마찬가지로 bid값을 제시하게 된다. bid값은 식 (3)과 같이 bid 상수( $C_{bid}$ )와 신뢰도( $S_r$ ) 그리고 발화 강도( $F$ )의 곱으로 표시한다.

$$Bid_r = C_{bid} \cdot S_r \cdot F, \quad (3)$$

분류자 시스템과 같이 bid-process를 통해서 제어행동을 하기 위한 권한을 얻기 위해 정합된 분류자 들은 bid값을 제시하고, 이 값에 확률적으로 비례해서  $N$ 개의 규칙을 선택하여 rule-base를 구성한다. 일반적인 Bucket Brigade 알고리즘에서는  $t+1$  시간에 활성화 되는 분류자는  $t$  시간의 분류자에 의해 생성된 메시지에 의해서 활성화된다. 그러므로 현재 만족된 분류자들은 현재 분류자를 만족하는 메시지를 생성한 이전 분류자들에게 대가를 지불하게 된다. 반면 퍼지 분류자 시스템에서는 규칙과 규칙사이의 직접적인 영향

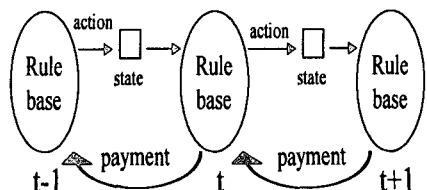


그림 4. 내재적 Bucket Brigade 알고리즘  
Fig. 4. Implicit Bucket Brigade Algorithm

을 설명하기가 부적절하다. 그림 4에서와 같이  $t$  스텝의 rule-base는  $t-1$  스텝의 rule-base에 의한 제어행동에 의해서 만들어진 상태를 만족하는 rule-base가 된다. 이것은  $t$  스텝의 rule-base가  $t-1$  스텝의 rule-base에 의해서 간접적으로 영향을 받아서 활성화되었고 볼 수 있다. 이것을 내재적인 Bucket Brigade 알고리즘이라고 한다[8].

내재적인 Bucket Brigade 알고리즘을 퍼지 분류자 시스템에 적용하기 위한 수식은 아래와 같다. 우선 식 (4)와 같이  $t$  스텝 rule-base를 구성한 모든 분류자들이 제시한 모든 bid값의 합을 구한다.

$$S_{bid}(t) = \sum_{k \in M(t)} Bid_k \quad (4)$$

여기서  $M(t)$ 는  $t$  스텝에 rule-base를 구성한 분류자의 인덱스 집합을  $k$ 는 분류자의 인덱스를 나타낸다. 그리고 시스템은 이 bid합을 이전 rule-base를 구성한 분류자들에게 분배한다. Bid-process 동안  $t$  스텝의 rule-base내의 규칙  $r$ 은  $t+1$  스텝에 신뢰도를 변화하게 된다.  $t+1$  시간에 규칙  $r$ 은 식 (3)에 의한 bid값을 신뢰도에서 제거한다. 그리고 규칙  $r$ 은  $t+1$  스텝에 얻어진  $S_{bid}(t+1)$ 를 규칙  $r$ 의  $t$  스텝 발화 강도에 비례해서 받게 된다. 이를 식으로 나타내면 식(5)와 같다.

$$S_r(t+1) = S_r(t) - Bid_r + \frac{F_r}{\sum_{i \in M(t)} F_i} S_{bid}(t+1) \quad (5)$$

### 2.4 지연 강화 학습

$t$  스텝에 구성된 rule-base의 규칙들은 각각 가중치를 가지는 출력을 만들고 제시된 출력들을 규합하여  $t$  스텝의 실수 값의 출력을 결정한다. 퍼지 분류자 시스템을 자율이동 로봇에 적용할 때, 시스템의 출력 값은 각 모터의 속도가 될 것이다. 로봇의 제어 스텝의 간격을 고려할 때, 시스템의 한 출력에 대한 행동이 로봇의 전체적인 행동에 크게 영향을 못 미칠 수가 있다. 제어 스텝의 간격을 길게 설정하면 한 출력에 대해서 많은 행동을 하여, 한 행동에 대해서 주변 상태가 너무 급격하게 변하기 때문에 학습의 성능이 떨어질 수가 있다. 반면 간격을 짧게 주면 환경에 대한 충분한 정보를 얻을 수 있지만 행동이 미비하기 때문에 적절하게 강화신호를 주는 것에 무리가 있다. 그러므로 충분히 행동을 한 후에 강화신호를 받아서 행동에 영향을 준 규칙들에게 분배를 한다.  $T$  스텝이 지난 후에 강화신호를 받아서  $T$  제어스텝 동안에 행동에 영향을 준 규칙들에게 분배한다. 현재  $t$  스텝에 강

화 신호를 받은 후  $T$  스텝이 지났다면 환경으로부터 강화신호  $R(t)$ 를 받게 된다. 그리고 식 (6)와 같이 영향을 미친 규칙들에게 행동에 기여한 정도에 따라서 강화신호를 분배한다.

$$S_r(t) = S_r(t+1) + \frac{\sum_{p \in L(t)} F_r(p)}{\sum_{p \in L(t), q \in B(t)} F_q(p)} R(t) \quad (6)$$

여기서  $L(t)$ 와  $B(t)$ 는 각각  $t$  스텝에서 과거  $T$  스텝 동안 입력 상태집합과 rule-base에 참여한 규칙 집합을 나타내고  $F_r(p)$ 는 규칙  $r$ 이  $p$  상태일 때 규칙의 발화 강도를 나타낸다. 이때  $T$  스텝 이내에 특정한 조건을 만족을 하면 강화신호를 준다. 예를 들면 자율이동로봇이 먹이를 먹었거나 목표지점에 도달했을 경우이다. 분류자 시스템에서 사용되지 않는 분류자를 찾기 위해서 tax를 각각 분류자에게 부과한다. 퍼지 분류자 시스템에서도  $T$  스텝 이 지나고 강화신호를 받은 시점에서 분류자 리스트에 존재하는 모든 분류자에게 식(7)과 같이 tax를 부과한다.

$$S_r(t) = S_r(t-1) - C_{tax} S_r(t-1) \quad (7)$$

여기서  $C_{tax}$ 는 Tax constant이고  $C_{tax} \ll C_{bid}$ 이다.

### 2.5 규칙생성

어느 정도의 제어 스텝이 지나도 시스템 성능의 향상이 없으면 퍼지 분류자 리스트의 분류자들이 가지고 있는 신뢰도 값을 GA의 적합도로 사용하여 새로운 규칙생성을 시도한다. 분류자 리스트에 입력 상태를 만족하는 규칙이 존재하지 않을 때는 cover detector[9] 방법을 사용하여 입력을 만족하는 규칙을 추가한다.

분류자 시스템에서는 가장 우수한 하나의 규칙을 찾는 것이 아니라, 가장 적합한 규칙의 집합을 찾는 것을 목적으로 한다. 이를 위해서 그림 5의 crowding replacement 방법을 적용한다. 룰렛 선택과 교차, 돌연

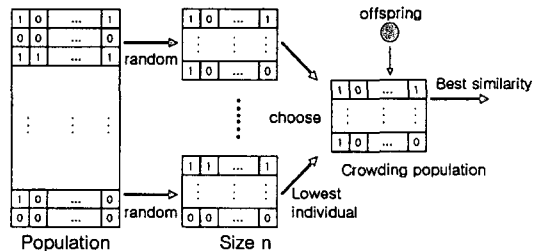


그림 5. 집단 대체 방법  
Fig. 5. Crowding Replacement

변이 연산을 통해서 만들어진 새로운 규칙을 어떻게 분류자 리스트에 포함시킬 것인가 하는 문제이다. 전체 분류자 리스트에서 규칙을  $n$ 개씩 임의적으로 선택한다. 선택된 규칙 중 신뢰도 값이 가장 낮은 규칙만을 따로 선택한다. 이 과정을 crowding population을 채울 때까지 반복한다. 선택된 crowding population중에서 새로 만들어진 규칙과 가장 유사한 규칙을 서로 대체하는 방법을 사용한다[1].

### 3. 실험

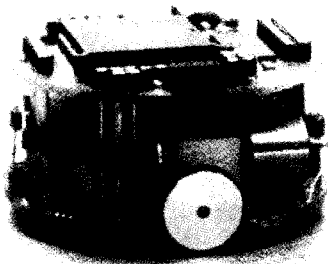
본 논문에서 제안된 퍼지 분류자 시스템을 자율이동 로봇의 충돌회피 학습에 적용하여 그 유용성을 확인하였다. 실험에 사용한 자율이동로봇인 Khepera(그림 6(a))는 그림 6(b)와 같이  $\pm 30^\circ$  범위를 감지할 수 있는 8개의 적외선 센서를 이용하여 장애물과의 거리 정보를 얻는다. 실험에서는 그림(b)에서 0~5번까지 6개의 센서를 2개씩 그룹을 지어 사용하였다. 예를 들어 센서 0과 센서 1은 센서 S0라는 하나의 그룹을 형성하여  $\pm 60^\circ$  범위의 외부 환경에 대한 정보를 얻는다. 시스템의 출력 값은 자율이동로봇의 두 모

터를 구동하는데 사용하게 된다.

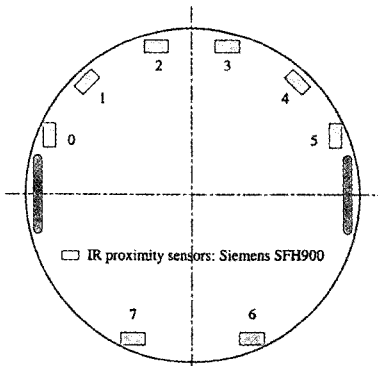
표 1은 입출력 변수의 범위를 나타내고 있다. 입력 변수는 외부 환경에 대한 센서 값을 의미한다. 장애물이 자율이동로봇과 가까운 곳에 있으면 입력 변수의 값이 커지고, 먼 곳에 있으면 입력 변수의 값이 작아지게 된다. 만약 입력 변수의 값이 1023이면 로봇이 장애물과 충돌한 경우이다. 출력 변수는 모터의 속도를 결정하게 된다. 출력 변수는 모터의 속도를 의미하며, 이 값이 양수이면 모터가 전진하고, 음수인 경우 후진한다. 그림 7은 각각 5개의 소속함수로 구성되어 있는 입력변수와 출력변수의 퍼지 집합을 나타낸다.

자율이동로봇은 센서를 이용하여 외부 환경에 대한 정보, 즉 장애물과 로봇과의 거리 정보를 얻어내고 이를 이용하여 장애물과 충돌하지 않고 주행할 수 있는 적절한 퍼지 규칙을 학습하게 된다. 각각의 입력변수와 출력변수는 5개의 소속함수가 존재하므로, 3개의 입력 변수와 2개의 출력 변수의 소속함수로 나타낼 수 있는 규칙은 3125개의 경우의 수를 갖는다. 즉 퍼지 분류자 시스템은 이 3125개의 탐색영역 후보 중에서 장애물과 충돌하지 않고 이동하기 위한 적절한 퍼지 규칙의 집합을 찾는다.

자율이동로봇이 이동할 수 있는 공간은 장애물이 존재하는 한 변이 40 cm인 정사각형인 공간으로 설정하였다. 자율이동로봇이 센서로부터 외부환경에 대한 정보를 얻고, 퍼지 분류자 시스템에 의해 생성되는 출력 값으로 한번 이동하는 것을 1 스텝이라 한다. 150 스텝을 1회 실행으로 하고, 매회 실행 후 로봇을 원래 위치로 이동시키고 정사각형 공간내의 장애물의 위치



(a) Khepera 로봇



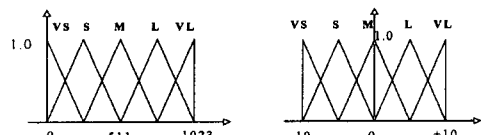
(b) 로봇의 센서 구성

그림 6. 자율이동로봇의 외형과 센서 구성  
Fig. 6. AMR and sensor configuration

표 1. 입출력 변수값의 범위  
Table 1. Range of input/output variable

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 0.05 mm.  
단위 값 당 모터의 속도: 0.08 mm/10 ms.



(a) 입력 변수의 소속 함수

(b) 출력 변수의 소속 함수

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

그림 7. 입출력 변수의 소속 함수

Fig. 7. Membership function for input/output variable

표 2. 유용한 퍼지 규칙 집합  
Table 2. A useful set of rules

	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	S	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	L	M
17	M	M	VS	VL	M

와 모양을 바꾼다. 5 스텝 후 강화 신호를 받고, 강화 신호를 10번 받은 후 규칙 생성을 시도한다. 주요 파라미터 값은 초기 분류자의 개수를 100개, 최대 스텝을 150 스텝, 실행횟수를 1000회, 분류자의 초기 신뢰도를 100, Bid 상수 값을 0.1, Tax 상수 값을 0.01, 교차확률을 0.8, 돌연변이 확률은 0.01로 설정하였다. 강화신호는 식 (8)과 같이 설정하였다.

$$R(t) = \alpha D(t) - \beta CO(t) \tag{8}$$

여기서  $D(t)$ 는 이동거리를,  $CO(t)$ 는 충돌 횟수를 나타낸다.

여러 번의 실험 결과 약 300여 회 실행 후 자율이동로봇이 장애물과 충돌하지 않고 이동하는 행동을 보여 줬다. 각 실험마다 얻어지는 유용한 퍼지 규칙의 집합은 조금씩 달랐지만, 로봇은 유사한 행동을 하면서 장애물을 회피하였다. 실험 후 시스템에는 많은 퍼지 규칙들이 존재하지만, 퍼지 규칙들의 신뢰도를 토대로 이들 중 10~15개 정도의 퍼지 규칙들만이 장애물과 충돌하지 않고 이동하는 행동에 사용됨을 확인할 수 있었다. 실험 결과 중 한가지를 표 2의 유용한 퍼지 규칙 집합에 나타내었다.

#### 4. 결 론

학습과 진화의 기법을 접목하여 인간의 사고능력을

모델링한 분류자 시스템은 실시간 학습방법으로 주목을 받고 있다. 하지만 이산적인 정보만을 다루는 분류자 시스템에서의 정보의 손실은 중요한 문제로 생각되어 졌다. 그러나, 퍼지 분류자 시스템은 분류자 시스템이 연속적인 입력 값을 연속적인 출력 값으로 매핑 할 수 있도록 해준다. 본 논문에서 퍼지 분류자 시스템은미시간 방법을 사용하고 입력 값을 퍼지화 하여 메시지를 만들어낸다. 시스템은 퍼지화 된 메시지들을 메시지 리스트에 저장하고 내재적인 Bucket Brigade 알고리즘을 사용하여 분류자의 신뢰도를 수정한다. 또한, 시스템은 주위 환경에 따라 강화 신호가 결정되고, 이 신호를 바탕으로 강화 학습을 수행한다. 시스템의 성능 향상이 없으면 새로운 규칙 생성을 시도한다. 이런 과정을 통하여 퍼지 분류자 시스템은 하나의 규칙을 찾는 것이 아니라 환경에 적응하는 규칙의 집합을 찾아내게 된다. 제안된 퍼지 분류자 시스템을 자율이동로봇의 장애물 회피 학습에 적용하여 환경에 적합한 퍼지 규칙 집합을 찾을 수 있음을 확인하였다.

#### 참고문헌

- [1] David E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989.
- [2] Holland J. H., "Properties of the bucket brigade algorithm," Proceedings of the First International Conference on Genetic Algorithms, pp. 1-7, 1985.
- [3] Whitehead, S. D., & D. H. Ballard, "Learning to perceive and act by trial and error," Machine Learning, 7, pp. 45-83.
- [4] Valenzuela-Rendon M, "The Fuzzy classifier system: Motivations and first results," Parallel Problem Solving from Nature-PPSNII, Springer-Verlag, pp. 330-334, 1991.
- [5] Parodi A., Bonelli P., "A new approach to fuzzy classifier systems," Proceedings of the Fifth International Conference on Genetic Algorithms, pp. 223-230, 1993.
- [6] Furuhashi T., Nakaoka K., Morikawa K., Uchikawa Y., "Controlling excessive fuzziness in a fuzzy classifier system," Proceeding of the Fifth International Conference on Genetic Algorithms, pp. 635, 1993.
- [7] 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.
- [8] 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.

- [9] Wilson, S. W., "Knowledge growth in an artificial animal," Proceedings of the first International Conference on Genetic Algorithms and their Applications, 1985.



**반 창 봉 (Chang-Bong Ban)**

2000년 : 중앙대학교 전자전기공학부  
학사

2000년~현재 : 중앙대학교 전자전기공  
학부 석사과정

관심분야 : 진화하는 하드웨어, 지능로  
봇시스템 등



**심 귀 보 (Kwee-Bo Sim)**

1984년 : 중앙대학교 전자공학과 학사

1986년 : 중앙대학교 전자공학과 석사

1990년 : The University of Tokyo 전  
자공학과 박사

1990년 : 동경대학 생산기술연구소 연  
구원

1998년~현재 : 한국 퍼지 및 지능시스  
템 학회 논문지 편집위원장 및 이사

1999년~현재 : 한국 뇌학회 학술위원

1991년~현재 : 중앙대학교 전자전기공학부 교수

관심 분야 : 인공생명, 진화연산, 지능로봇시스템, 뉴로 퍼지 및  
소프트 컴퓨팅, 자율분산시스템, 로봇비전, 진화하  
는 하드웨어, 인공면역계 등