
Fusion Method와 유전자 알고리즘을 이용한 퍼지 제어 시스템의 설계

이영신*, 이윤배**, 나영남***

A Design of Fuzzy Control System Using Fusion Method and Genetic Algorithm

Young-Sin Lee, Yun-Bae Lee, Young-Nam Na

본 연구는 '98년도 조선대학교 연구비 지원에 의해 수행됨.

요 약

퍼지 제어기는 퍼지 제어기 설계를 위한 소속 함수와 제어규칙의 구성 시 전문가의 경험적인 지식에 전적으로 의존하고 있다. 때문에, 제어기의 객관성이 보장되지 않으며 설계자가 예상하지 못한 플랜트 매개 변수의 변동이나 돌발적인 상황에서는 제어기의 성능이 저하되는 문제점을 가지고 있다. 이와 같은 문제점을 개선하기 위해서, 본 논문에서는 제어 규칙의 수를 감소시키고 직관적인 제어기를 구성하기 위하여 Fusion Method과 퍼지 제어기의 제어 규칙과 소속 함수의 매개변수를 조정하여 효율성을 향상시킬 수 있는 유전자 알고리즘을 적용한 자기 조직화가 가능한 퍼지 제어기를 제안하였다. 제안된 알고리즘의 타당성 검증을 위하여 무인 운송 장치 제어기를 구성하여 다양한 환경에서 모의 실험을 수행하였다.

Abstract

A fuzzy controller need membership functions and the control rules depend on heuristic knowledge of expertises entirely. On account of, it is possible that a desired performance of a fuzzy controller can not be

* 서강정보대학 생활정보과 전임강사

** 조선대학교 컴퓨터공학부 교수

*** 조선이공대학 정보통신과 전임강사

접수일자 : 1999년 12월 31일

guaranteed or easily degraded under some circumstances such as a change of plant parameter which experts do not considered. Therefore, in this paper we tried to increase the controller's efficiency by adjusting the control rules and the parameters of the membership functions by using a genetic algorithm. We also proposed the Self-Organizing Fuzzy Controller which uses the Fusion Method in order to minimize the number of control rules and to construct the intuitive controller. For validation of the proposed algorithm, we design the Autonomous Guided Vehicle Controller, then apply to variant condition.

I. Introduction

퍼지 제어기는 기존의 선형제어기 보다 많은 이점을 가지고 있는데 언어적인 표현을 사용함으로써 이해가 쉽고, 불명확하고 모호한 환경에서 지식의 표현을 쉽게 할 수 있다. 따라서, 미지의 프로세스나 복잡한 시스템의 수학적 모델링이 불가능한 경우에도 퍼지 추론에 의하여 적절한 제어량을 얻을 수 있으며 비선형적인 시스템에도 적용이 가능하다.

그러나, 퍼지 제어기 설계를 위한 소속 함수와 제어규칙의 구성은 전문가의 경험적이고 발견적인 지식에 전적으로 의존하고 있다. 때문에, 전문가의 능력에 따라 제어기의 성능이 결정되어 제어기의 객관성이 보장되지 않으며 설계자가 예상하지 못한 플랜트 매개 변수의 변동이나 돌발적인 상황에서는 제어기의 성능이 저하되는 문제점을 갖고 있다. 그리고, 변화하는 환경에 능동적으로 대처할 수 있는 능력이 결여되어 있다.^{3,4,5)}

본 논문에서는 이와 같은 점을 개선하기 위하여, 유전자 알고리즘을 사용하여 학습과 적응이 가능한 자기 조직화 퍼지 제어기(Self-Organizing Fuzzy Controller : SOFC)를 구성한다. 또한, 제어 규칙의 수를 감소시키고 직관적인 제어기를 구성하기 위하여 Fusion Method를 사용한다.

II. Fuzzy Logic Controller

퍼지 시스템의 응용 영역이 커지고 그 응용 영역의 입출력 변수가 많아질수록 전문가라 할지라도 그 변수들 간의 관계를 명확히 파악하는 것이 어렵다. 따라서, 지식 획득 문제가 퍼지 시스템의 이점을 위해 가장 큰 문제로 대두되고 있다. 이에

관한 대표적인 연구로써 퍼지 시스템과 신경망의 학습 능력을 결합하는 등의 다양한 방법들이 제안되고 있다.^{6,7,8)} 또한, 퍼지 시스템과 유전자 알고리즘을 결합한 연구도 다양하게 진행되고 있다.^{9,10)}

이러한 시도는 시스템의 성능 향상을 추구하고, 시스템의 입출력 데이터를 사용한 직접적인 퍼지 시스템 모델링을 목표로 하고 있다. 퍼지 시스템의 또 다른 어려운 점은 적절한 소속 함수를 어떻게 정의하느냐의 문제이다.

2.1 입력 변수와 퍼지 값 결정

집합의 의미는 소속 함수에 의해 특성 지어진다. 퍼지 값들의 소속 함수 형태로 결론부 (THEN-part)의 wi는 crisp한 단일 값을 갖는 것으로 하고, 조건부(IF-part)의 Aj, Bk들의 소속 함수는 삼각형 모양을 갖는 것으로 한다.

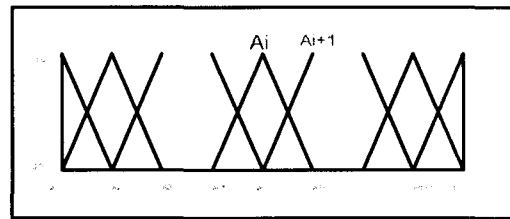


그림 1. 조건부 퍼지 제어 규칙의 삼각형 소속 함수
Fig. 1 The triangular membership function of premise part of fuzzy control rules

그림 1의 퍼지 집합 Ai의 소속 함수를 수식으로 나타내면 다음과 같다.

$$\mu_{A_i} = \begin{cases} (x - a_{j-1}) / (a_j - a_{j-1}), & \text{만일 } a_{j-1} \leq x \leq a_j \\ (-x + a_{j+1}) / (a_{j+1} - a_j), & \text{만일 } a_j \leq x \leq a_{j+1} \\ 0, & \text{otherwise} \end{cases} \quad (2.1)$$

퍼지 변수의 퍼지 값들이 동일한 형태의 삼각형 소속함수를 갖는 경우, 몇 가지 특성을 보면, 대집합(Universe of discourse)을 $U=\{x_{min}, x_{max}\}$ 라 하고, m 개의 소속 함수로 구분할 때,

모든 $x \in U$ 에 대해, $\sum_{i=0}^m \mu_{A_i}(x) = 1$ 이며, 모든 $x \in [a_j, a_{j+1}]$ 에 대해, $\mu_{A_j}(x) = 1 - \mu_{A_{j+1}}(x)$ 인 성질을 갖는다.

퍼지 제어를 위해 사용되는 퍼지 추론은 다음과 같은 형태의 다 입력 단일 출력 형태(multi-input single-output: MISO)의 규칙들의 집합으로 표현할 수 있다.

$$R_N: \text{IF } X \text{ is } A(n) \text{ and } Y \text{ is } B(n) \text{ THEN } Z \text{ is } C(n) \dots\dots\dots (2.2)$$

여기서, $A(n)$, $B(n)$, $C(n)$ 은 각각 입력 변수 X , Y 와 출력변수 Z 에 대한 언어 항들의 퍼지 집합이고, 상기 식 (2.2)는 다음과 같은 퍼지 관계(fuzzy relation)로 변환된다.

$$R = \text{also}(R_1, R_2, \dots R_N) \dots\dots\dots (2.3)$$

조건부에 사용된 2개의 입력변수는 각각 자율운송장치(Autonomous Guided Vehicle : AGV)가 목표 선으로부터 떨어진 거리 d 와 진행 방향 θ 를 각각 나타낸다. 결론부에 사용한 출력변수는 AGV의 핸들링 각도를 나타낸다. 입력과 출력 변수에 대해 다음과 같이 퍼지 집합을 설정하였다.

$$D \in \{ LF(\text{Left Far}), LN(\text{Left Near}), MD(\text{Middle}), RN(\text{Right Near}), RF(\text{Right Far}) \}$$

D 는 AGV와 목표선과의 거리를 나타내는 집합이다. 퍼지 변수 LF 는 AGV가 목표 선에서 왼쪽으로 멀리 벗어나 있음을 나타낸다. MD 는 목표선 위를 주행하고 있음을 나타내며 RN 은 목표선의 오른쪽 가까이 주행하고 있음을 나타낸다.

그림 2,3,4와 같이 입력과 출력 변수에 각각 퍼지 집합을 설정하였다. 일반적으로 소속 함수는 삼각형, 사다리꼴, 종형, 지수형 등이 있는데 본 논문에서는 일반적으로 많이 사용되고 계산이 편리한 삼각형의 소속 함수를 사용하였다.

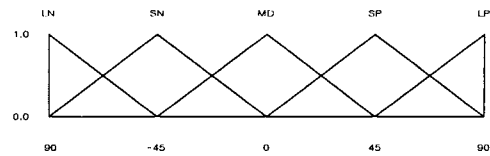


그림 2. AGV의 움직이는 방향 소속 함수
Fig. 2 The membership function for the moving direction of AGV

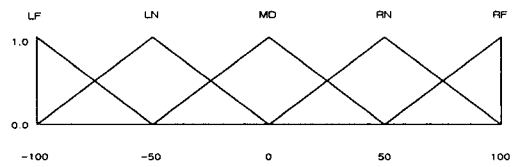


그림 3. 목표 경로와의 거리 소속 함수
Fig. 3 The membership function for distance from target path

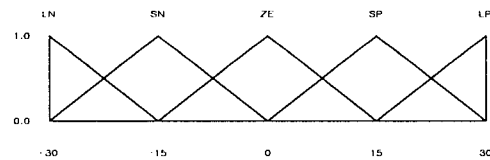


그림 4. AGV의 핸들링 소속 함수
Fig. 4 The membership function for the handling angle of AGV

2. 퍼지 제어 규칙

퍼지 제어기 설계에 사용되고 있는 퍼지 제어 규칙은 특성상 상태 평가형(state evaluation)과 목적 평가형(object evaluation)이 있다. 본 논문에서는 일반적으로 사용되고 있는 상태 평가형 제어 규칙을 사용하여 모의실험 하였는데 형식은 다음과 같다.

$$R_i : \text{If } x \text{ is } A_i, \dots, \text{ and } y \text{ is } B_i \text{ then } z = f_i(x, \dots, y) \dots\dots\dots (2.4)$$

따라서 본 논문에서는 우선 경험적 제어규칙을 부여한 후, 유전자 알고리즘을 사용한 적응제어(Self-Organizing Control : SOC)가 가능한 퍼지 제어기에 의하여 규칙의 생성 및 재구성이 가능하도록

록 한다. <표 1>은 초기 값으로 사용된 퍼지 제어 규칙을 나타낸 것이다.

<표 1> 퍼지 제어 규칙

<Table 1> Fuzzy Control Rules

거리 방향	LF	LN	MD	RN	RF
LN	LP	LP	LP	SP	ZE
SN	LP	SP	SP	ZE	SN
MD	LP	SP	ZE	SN	LN
SP	SP	ZE	SN	SN	LN
LP	ZE	SN	LN	LN	LN

2.3 퍼지 추론, 비퍼지화, command fusion

본 논문에서는 퍼지 추론 과정을 Command Fusion 방법을 사용하였다. 퍼지 추론을 통해 얻어진 최종 값은 플랜트를 제어하기 위한 결정 값이 아니므로 플랜트의 입력으로 사용하기 위해서는 비퍼지화 해야 한다. 비퍼지화의 주된 방법으로 (1) the Mean of Maximum (MOM)와 (2) the Center of Area (COA) 방법이 있다. MOM 비퍼지화 방법은 퍼지 명령에서 최고의 소속 정도를 갖는 값들의 평균을 계산하는 것이다. COA 비퍼지화 방법은 전체 퍼지 명령의 무게 중심을 계산하는데, COG(Center Of Gravity), Simplified COG, Max Criterion, 그리고, Mean of Maxima 방법등이 있다.

여기서, 비퍼지화 방법중 COG는 대집합 U를 양자화 한 후의 비퍼지화 값 u^* 를 다음과 같이 정의한다.

$$u^* = \text{defuzzifier}(B') = \frac{\sum_{j=1}^n \mu_{B'}(u_j) \cdot u_j}{\sum_{j=1}^n \mu_{B'}(u_j)} \dots\dots\dots (2.5)$$

그러나, COA 비퍼지화 방법을 AGV 제어에 직접 적용하는데는 문제가 있다. 그림 5는 정면에 장애물이 있는 상태에서 생성된 소속 함수를 보인 것이다.

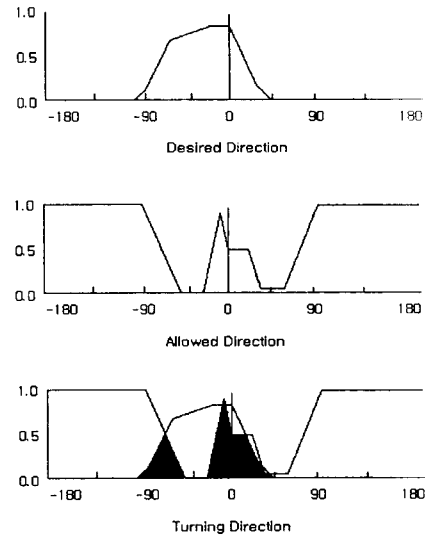


그림 5. Command Fusion의 예
Fig. 5 An Example of Command Fusion

즉, 경로 추적기와 방해물 회피기에 의해 생성된 진행 가능 방향 퍼지 집합과 진행 금지 방향 퍼지 집합을 조합하여 산출한 퍼지 연산 결과를 나타내고 있다. 검은 색 부분이 추론 결과로서 얻어진 AGV의 진행 예정 방향을 의미한다. AGV의 진행 방향과 조합된 퍼지 명령은 두 개의 최고점을 갖는 소속 함수를 갖게된다. 이 두 개의 최고점을 갖는 퍼지 집합에 COA 비퍼지화 기법을 적용하는 것은 AGV가 장애물에 더 가까이 접근하는 결과를 가져온다. 일반적으로 COA 방법은 금지된 정보를 포함하는 곳에 적용할 때 문제를 일으킬 수 있다.

따라서, 본 논문에서는 CLA(Centroid of Largest Area)라 불리는 변형된 새로운 비퍼지화 방법을 제안하였다. 이것은 가장 큰 영역의 퍼지 부분 집합만을 선택, COA 방법을 이용하여 비 퍼지화 하는 것이다.

2.4 퍼지 논리에 기반을 둔 동작기 구조

다른 동작기에 관한 연구와 마찬가지로, 본 논문의 AGV 제어 시스템도 다수의 동작기로 구성된 구조를 가지며, 퍼지 논리를 기반으로 한다. 각 동

작기는 AGV의 센서 데이터, 상태 데이터와 경로 정보에 관한 것 등과 관계된다. 이들은 그림 6과 같이 2개의 동작기인 경로 추적기와 방해물 회피기가 결합된 구조를 가지고 있다. 다중 동작기는 일반적으로 퍼지 추론 모듈을 공유할 수 있다. 퍼지 제어 명령은 모든 동작기에 의해 생성되고 조합되며 최종적으로 기계적으로 제어가 가능한 형태의 제어 명령을 생성하기 위하여 비 퍼지화를 행한다. 위와 같은 구조에 기반을 두고, AGV를 실험하였다.

기본 알고리즘은 다음의 네 단계로 구성된 구조에 의해서 모든 제어 사이클마다 수행된다.

- (1) 경로 추적기가 원하는 진행 방향을 결정한다.
- (2) 방해물 회피기는 금지된 진행 방향을 결정한다.
- (3) Command Fusion 모듈은 원하는 방향과 금지된 방향을 조합한다.
- (4) 조합된 퍼지 명령은 비퍼지화에 의해 crisp 형태의 이산적인 명령으로 변환된다.

2.4.1 경로 추적기

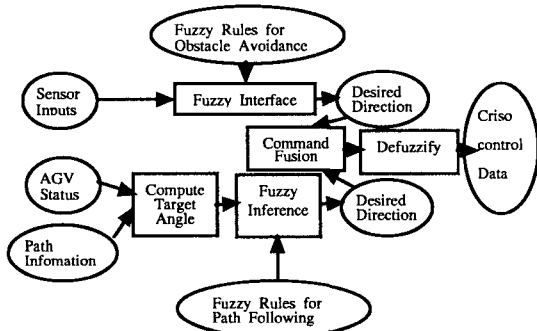


그림 6. 퍼지 논리 AGV 운행 제어기
Fig. 6 A Fuzzy logic Based AGV Navigation Controller

경로 추적기는 AGV의 현재 위치를 기반으로 다음과 같이 세 단계를 거쳐 원하는 진행 방향을 결정한다.

첫째, AGV에 가장 근접한 경로상의 점으로부터 거리 d 를 예측하기 위하여 전방의 가장 근접한 목표가 되는 위치를 목표점으로 정한다. 둘째, 동작

기는 AGV의 현재 진행 방향과 현재 위치에서 목표점까지의 벡터사이의 각도인 목표 각도 θ 를 계산한다. 그림 7은 이와 같은 상태를 보여주는 것으로 AGV의 진행 방향이 북쪽을 향하고 있을 때 목표 각도 θ 는 좌측 방향으로 약 30도 정도이다. 셋째, 동작기는 수치적으로 명확하게 표현된 목표 각도를 퍼지 형태의 원하는 방향으로 확장시켜 변환하기 위하여 퍼지 규칙의 집합을 사용하며, 이때문에 AGV는 경로를 따라가는 동안에 더 유연하게 장애물 회피를 할 수 있게 된다.

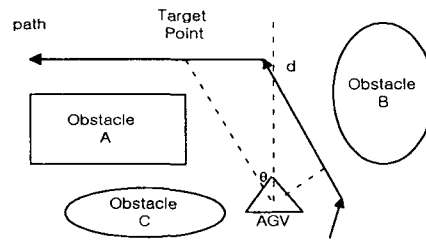


그림 7. 장애물 회피의 예
Fig. 7 An Example of obstacle avoidance

2.4.2 장애물 회피

본 논문에서 제시되는 AGV의 방해물 회피기는 진행 금지된 방향을 표시하는 퍼지 집합을 생성하기 위해 초음파 센서를 사용한다고 가정한다. AGV에 부착된 초음파 센서는 그림 8과 같이 장착되어 있다는 가정 하에 실험을 수행하였다.

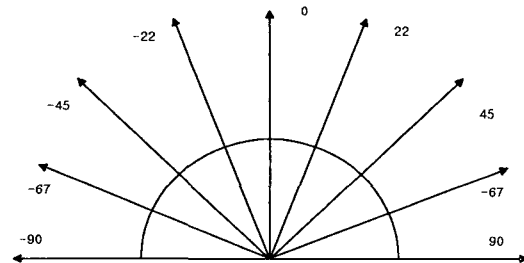


그림 8. 센서의 구조
Fig. 8 The structure of sensors

동작기는 먼저 각각의 초음파의 입력을 비교하고, 센서에 의해 감지된 가장 근접한 장애물과의

거리를 측정하고, 그 센서에 장애물과의 거리를 나타내는 퍼지 집합 Near를 부여한다. 비교의 결과에 기반을 두고, 동작기는 진행 불가로 추정되는 각도의 센서를 결정한다. 각각의 센서 데이터는 이웃하는 센서들과 부분적으로 겹쳐지게 되는데, 그것을 센서의 방향에 영향을 받는 연관된 진행 금지 방향의 소속 함수로 설계하였다. 단지 장애물 회피기와 연관된 모든 퍼지 규칙은 동작을 시작하고 그들의 퍼지 결론은 MAX 연산자를 사용하여 조합된다. 그림 9는 전술한 AGV의 예시된 상태인 <표 2>와 같은 상태에 기반을 두고 센서 입력을 결합한 예를 보인 것이다.

동작기의 퍼지 추론 모델은 t-conorm 연산자 대신 MAX 연산자를 사용한다.

장애물 회피의 관점에서, 전방 센서에 의해 감지된 거리 d의 장애물은, 좌측 센서에 의해서 감지된 장애물 보다 AGV에 더 근접한 것으로 간주한다. AGV 운행 제어기에는 원하는 목표 방향과 금지된 방향에 대한 퍼지 결론들이 조합된 것이다.

III. 유전자 알고리즘과 진화 프로그래밍

진화 프로그램은 유전자 알고리즘의 일반화로 인식될 수 있다. 또 다른 형태의 진화 기반 시스템은 Holland의 유전자 알고리즘이다. 진화 프로그램은 확장된 유전 연산자들과 함께 보다 풍부한 데이터 구조들을 사용하고 있으나, 고전적 유전자 알고리즘은 개체에 대하여 일정한 길이의 이진 스트링과 이진 돌연 변이와 이진 교배라는 두 연산자를 사용한다. 즉, 유전자 알고리즘의 구조는 진화 프로그램과 동일하다. 그러나 진화 프로그램은 염색체가 반드시 비트 스트링으로 표현 할 필요는 없으며, 변천과정은 주어진 구조와 문제에 적합한 다른 유전 연산자를 포함하고 있다. 유전자 알고리즘은 해결하고자 하는 문제에 대한 가능한 해들을 일정한 형태의 유전자라는 데이터구조로 표현한 다음 이들을 점차적으로 변형함으로써 자연계의 적자생존과 우성 유전과 같은 개념으로 점점 더 좋은 해들을 생성하게 하는 것이다. 현재 유전자 알고리즘은 탐색, 최적화 및 기계 학습의 도구로 많이 사용되고 있으며 구조가 단순하면서도 응용 범위가 넓

은 특징을 가지고 있다. 최근에는 여러 가지 이론과의 결합이 활발히 이루어지고 있다. 본 논문에서는 자기조직화가 가능한 퍼지제어기(SOFC)를 구성하기 위하여 진화프로그래밍을 적용하였다. 적용된 진화프로그램은 퍼지제어기의 매개변수들을 자기조정하여 퍼지제어기의 제어 효율성을 향상 시킬 것이다.

그림 9는 유전자 알고리즘 구조를 도해한 것이다.

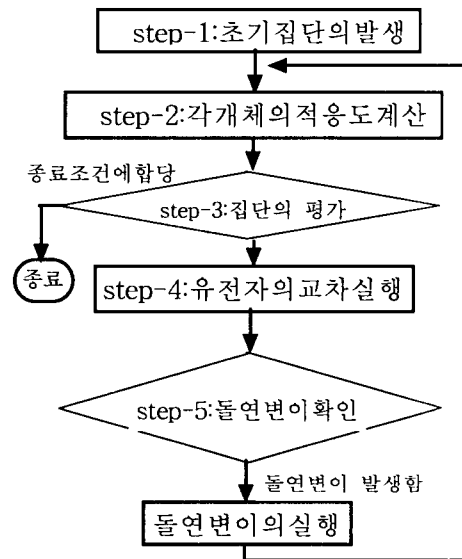


그림 9. 유전자 알고리즘의 구조
Fig. 9 The structure of Genetic Algorithm

유전자 알고리즘은, 적합도에 비례하는 룰렛 선택 사용한다. 룰렛 선택은 다음과 같이 구성된다.

- 각 염색체 $v_i(i=1, \dots, \text{pop_size})$ 에 대한 적합도 값 $\text{eval}(v_i)$ 를 계산한다.
- 개체 집단의 총 적합도는 다음 수식을 이용하여 계산한다.

$$F = \sum_{i=0}^{\text{pop_size}} \text{eval}(v_i) \dots\dots\dots (3.1)$$

각 염색체 $v_i (i=1, \dots, \text{pop_size})$ 에 대한 선택 확률 p_i 는 다음 수식을 이용하여 계산한다.

$$p_i = \text{eval}(v_i) / F \dots\dots\dots (3.2)$$

- 각 염색체 $v_i (i=1, \dots, \text{pop_size})$ 에 대한 누적

확률 q_i 는 다음 수식을 이용하여 계산한다.

$$q_i = \sum_{j=1}^i b_j \dots\dots\dots (3.3)$$

선택 과정은 룰렛 선택을 pop_size 번 만큼 반복 수행한다. 즉, 다음과 같은 방법으로 새로운 개체 집단을 위한 하나의 염색체를 선택한다.

- ◎ 범위 [0,1] 사이의 부동 소수점 난수 r 을 발생시킨 후, 만일 $r < q_i$ 이면 첫 번째 염색체(v_i)를 선택하고, 그렇지 않으면, $q_{i-1} < r \leq q_i$ 인 i 번째 염색체 v_i ($2 \leq i \leq pop_size$)을 선택한다.

1) 새로운 개체 집단의 각 염색체에 대하여:

- ▶ 범위 [0,1]사이의 부동소수점 난수 r 을 발생시킨 후, 만일 $r < p_c$ 이면 주어진 염색체를 선택하고 교배 연산을 적용한다.

선택된 염색체들을 임의로 교배시키는데, 결합된 염색체의 각 쌍에 대하여 범위 [1, $m-1$]로부터 교배점의 위치를 나타내는 임의의 정수 pos 를 발생시킨다.

여기서, (m 은 염색체의 총 길이).

두 개의 염색체

$(b_1 \ b_2 \ b_3 \ \dots \ b_{pos} \ b_{pos+1} \ \dots \ b_m)$ 과 $(c_1 \ c_2 \ c_3 \ \dots \ c_{pos} \ c_{pos+1} \ \dots \ c_m)$ 는 다음과 같은 자손의 쌍 형태로 대체된다.

$(b_1 \ b_2 \ b_3 \ \dots \ b_{pos} \ c_{pos+1} \ \dots \ c_m)$

$(c_1 \ c_2 \ c_3 \ \dots \ c_{pos} \ b_{pos+1} \ \dots \ b_m)$

그리고, 다음 연산인 돌연변이 연산자가 비트 별로 수행된다. 유전 연산자의 또다른 매개변수인 돌연변이 확률 p_m 을 통해 $p_m \cdot m \cdot pop_size$ 개의 비트가 돌연변이 되는, 즉 비트의 값이 반전될 기회가 동등하게 되는데 다음과 같은 돌연변이 연산을 수행한다.

2) 교배 연산후의 개체집단에 있는 각각의 염색체와 그 염색체 내의 각 비트에 대하여:

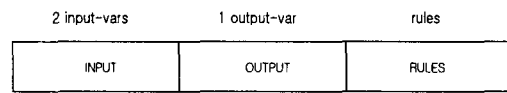
- ▶ 범위 [0,1] 내에서 난수 r 을 발생시킨 후, 만일 $r < p_m$ 이면 그 비트를 돌연변이 시킨다.

선택, 교배, 그리고 돌연변이를 한 후에 새로

운 개체 집단은 다음 평가를 받는다. 이 평가는 나중에 선택 과정을 위한 확률 분포를 만드는 데 사용된다.

3.1 유전자 표현형

AGV를 제어하는 퍼지 시스템의 매개변수를 유전자형으로 표현하고 이들 중 적합도가 높은 것들을 교배와 돌연변이를 시켜 새로운 유전자 코드를 생성한다. 이것을 다시 피드백하여 퍼지 시스템에 입력하는 방식으로 최적에 가까운 매개변수를 산출한다. 유전자형의 구조는 그림 10과 같다. 2개의 퍼지 입력 변수와 1개의 출력 변수와 제어 규칙을 인코드(encode)하고 있다.



<chromosome>

그림 10. 유전자의 구성
Fig. 10 Composition of gene

제어규칙을 구성하는 원소를 각각의 정수로 치환하여 미리 정해놓은 순서대로 나열함으로써 염색체 형태로 코딩 가능하다.

그림 11, 12은 각각 퍼지 소속 함수와 제어 규칙을 GA와 EP에 적합한 형태로 변환하는 방법을 예시한 것이다.

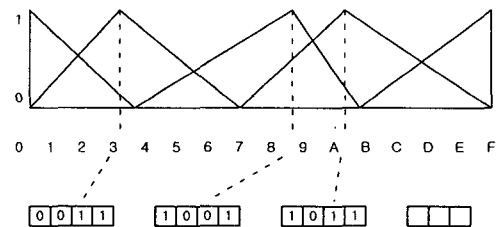


그림 11. 소속 함수의 코딩 방법
Fig. 11 The coding method of the membership function.

거리 \ 방향	LF	LN	MD	RN	RF
LN	0	0	0	1	2
SN	0	1	1	2	3
MD	0	1	2	3	4
SP	1	2	3	3	4
LP	2	3	4	4	4



00012 01123 01234 12334 23444

그림 12. 제어 규칙의 코딩 방법
Fig. 12 The coding method of control rules

3.2 GA를 이용한 퍼지 추론의 최적화

퍼지 소속 함수는 시스템에 따라서 여러 가지 형태를 적용할 수 있는데 본 논문에서는 가장 일반적으로 사용되는 삼각 함수를 퍼지 소속 함수로 사용한다.

다음은 소속 함수로 사용되는 삼각 함수를 수식화한 것이다.

$$\begin{aligned}
 f_i : x \rightarrow [0, 1] \\
 f_i(x, \alpha_i, \beta_i, \gamma_i) = 0, & \quad (x \leq \alpha_i) \\
 f_i(x, \alpha_i, \beta_i, \gamma_i) = (x - \alpha_i) / (\beta_i - \alpha_i), & \quad (\alpha_i < x \leq \beta_i) \\
 f_i(x, \alpha_i, \beta_i, \gamma_i) = (\beta_i - x) / (\beta_i - \gamma_i), & \quad (\beta_i < x \leq \gamma_i) \\
 f_i(x, \alpha_i, \beta_i, \gamma_i) = 0, & \quad (x \geq \gamma_i)
 \end{aligned}
 \tag{3.4}$$

여기서, β_i 는 삼각 퍼지 소속 함수 f_i 의 중점값, α_i 와 γ_i 는 양 끝점을 나타낸다. 본 논문에서는 다음과 같은 식에 의하여 위의 소속 함수를 조정한다.

$$\begin{aligned}
 \alpha_i &:= (\alpha_i + \delta sum_i) - \eta_i \\
 \beta_i &:= (\beta_i + \delta_i) \\
 \gamma_i &:= (\gamma_i + \delta_i) + \eta_i
 \end{aligned}
 \tag{3.5}$$

즉, 퍼지 소속 함수는 매개 변수 δ_i 와 η_i 에 의해 조정되며, δ_i 는 소속 함수를 좌우로 이동시키고 η_i 는 소속 함수의 폭을 조정한다. 본 논문에서는 퍼지 소속 함수를 최적화 해주는 매개 변수 δ_i

와 η_i 를 찾기 위해 GA를 사용하였다. 탐색체의 적합도를 계산하기 위한 평가 함수는 다음과 같다.

$$F = \frac{1}{(1 + \Delta e^2)} \tag{3.6}$$

위 식의 mean square error는 다음 수식으로 나타낸다.

$$\Delta e^2 = \sum_{i=1}^p (n_{ri} - n_{mi})^2 / \sum_{i=1}^p n_{ri}^2 \tag{3.7}$$

여기서, n_{ri} 는 퍼지 논리 제어기의 실제 출력 값이고, n_{mi} 는 기대 출력 값이다. v 는 이산화된 데이터의 수이다. 따라서 유전자 알고리즘의 각 세대는 mean square error가 감소하는 방향으로 진화하게 된다.

부의 언어항의 개수만큼 표현이 가능하므로 돌연변이가 일어나면 기존의 값을 제외한 나머지 중의 한 값을 새로운 제어 규칙으로써 갖게된다.

3.4 진화 프로그래밍

본 논문의 AGV 운행자는 진화 프로그램인 진화 운행자(Evolutionary Navigator : EN)를 사용한다. EN 알고리즘 전반부의 오프라인 경로 계획자는 출발점에서 목표점까지의 전역 경로를 탐색하며, 후반부의 온라인 계획자는 충돌이나 미지의 장애물 등을 피하기 위해 원래의 전역 경로 계획을 최적의 하위 이동(Subtour)으로 바꾸는 역할을 한다. 생성된 경로는 아래 그림 14와 같은 경로 노드들의 순서 리스트이다.

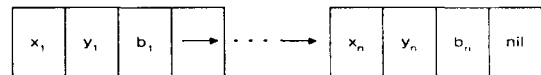


그림 13. 하나의 경로를 포함한 염색체
Fig. 13 The chromosome include one path

한 노드의 적합도를 조사하기 위한 방법은 상대적으로 간단한 Pavlidis의 알고리즘에 기반을 두고 있다. 한 염색체 $p = \langle m_1, m_2, m_3, \dots, m_n \rangle$ 에 대해서 전체 경로 비용으로 표시되는 적합도(fitness)는 다음과 같은 평가 함수에 의해 결정된다.

$$Path_{Cost(p)} = \sum_{i=0}^{n-1} cost(m_i) \dots\dots\dots (3.8)$$

여기서 $cost(m_i)$ 는 노드 m_i 의 비용이다. 이 비용은 만약 노드 m_i 가 적합하면 노드 m_i 로부터 다음 노드 m_{i+1} 까지의 길이를 의미하고, 아니면 하나의 상수 C 를 의미한다. 상수 C 는 부적합한 노드에 대한 벌칙을 나타내는 수이다. 진화 알고리즘에는 교배, 두 개의 돌연변이, 삽입, 삭제, 교체(swap) 등 모두 여섯 개의 연산자가 온라인, 오프라인에 포함되어 있다.

◎ 교배 : 유전자 알고리즘의 일점 교배와 유사한 것으로 선택된 두 개의 염색체가 절단되어 합쳐진다. 하지만 염색체의 교배점이 임의적(random)으로 선택되지는 않는다. 이 연산자가 사용될 확률은 p_1 이다.

▶ 돌연변이 1 : 염색체에 있는 노드들의 좌표 값을 미세 조정하는데 사용된다. 만약 한 염색체의 한 노드가 이 연산자를 위해 선택되었다면, 그의 좌표 값들이 바뀐다. 예를 들어, 좌표 $x \in \langle a, b \rangle$ 는 (좌표 y 역시) 다음과 같은 방법으로 바뀐다.

$$x' = \begin{cases} x - \alpha(t, x - a), & \text{만약 } \gamma = 0 \\ x + \alpha(t, b - x), & \text{만약 } \gamma = 1 \end{cases} \dots\dots\dots (3.9)$$

여기서 r = 랜덤 비트,
 t = 진화 과정에서의 현재 세대수
 $\delta(t, z) = 0$ 부터 z 사이의 값을 return.
 이 연산자는 비선형 최적화를 위해 비균일(non-uniform) 돌연변이를 기초로 모델화 되었으며, 사용될 확률은 p_2 이다.

▶ 돌연변이 2 : 하나의 값을 크게 변화시킬 때 유용하다. 어떤 장애물이 경로를 막고 있는 경우 온라인 계획 단계에서 종종 발생한다. 만약 한 염색체의 한 노드가 이 연산자를 위해 선택되었다면, 그의 좌표 값들이 바뀐다. 예를 들어, 좌표 $x \in \langle a, b \rangle$ 는 다음과 같은 방법으로 바뀐다.

$$x' = \begin{cases} x - \Delta(t, x - a), & \text{만약 } \gamma = 0 \\ x + \Delta(t, b - x), & \text{만약 } \gamma = 1 \end{cases} \dots\dots\dots (3.10)$$

여기서 함수 $\Delta(t, z)$ 는 0부터 z 사이의 값을

되돌려

IV. 실험 및 검증

4.1 실험 환경

모의 실험은 볼랜드 C++을 사용하여 수행하였다. 모의 수행은 세 가지 형태로 수행하였다. 실험은 경로를 따라 주행하는 성능과 장애물 회피에 중점을 두고 수행하였다.

본 논문에서 제안한 시스템의 전체적인 구조는 그림 14와 같다. 주행 성능을 평가하는 과정에서 d 는 오차 e 로써 표현되며 d 의 변화량 Δd 는 오차의 변화량 Δe 의 평가함수의 입력으로 사용하여 AGV의 주행 성능을 측정하였다. AGV의 주행 성능을 측정하기 위하여 다음 수식을 이용 Δt 마다 오차의 변이를 계산하여 누적 오차를 계산하였다.

$$\Delta E_{sum}(t) = E(t - \Delta t) + \frac{1}{2} \Delta E_{sum}(t - \Delta t) \dots\dots\dots (4.1)$$

여기서, $E(t - \Delta t)$ = t 와 Δt 사이의 오차의 변화량
 $\Delta E_{sum}(t - \Delta t)$ = 누적 오차의 변화량

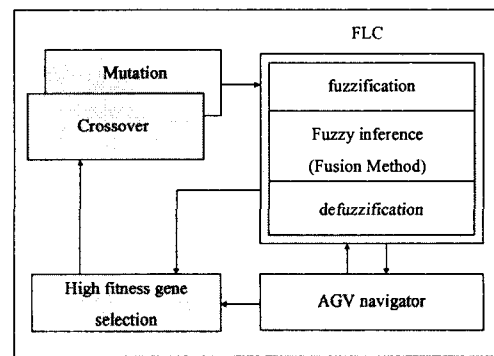


그림 14. GA에 의한 퍼지 시스템의 조정 구조
 Fig. 14 The tuning structure of the fuzzy system by GA

모의 수행에 사용된 GA 적용 환경은 다음과 같다.

· pop_size = 20

· Fitness 함수 : $Fitness_i = \frac{1}{1 + \sum_{i=0}^{216} |error_i|}$

· Error 값 : $error_i = E_i - F_i$

· 재생산 방식 : Roulette 방식,

$$p_i = \frac{Fitness_i}{\sum_{j=0}^{100} Fitness_j}, (\rho \text{는 reproduction 확률}) \cdot$$

maximum fitness : 0.99

· $p_c = 0.25, p_m = 0.01$

4.2 모의 실험

본 논문에서 제안된 퍼지 제어기를 검증하기 위하여, 그림 15,16,17과 같은 임의의 환경을 설정하여 주행 모의 실험을 하였다.

그림 15은 생성된 전역 경로를 따라 AGV가 주행한 경로를 표시한 것이다.

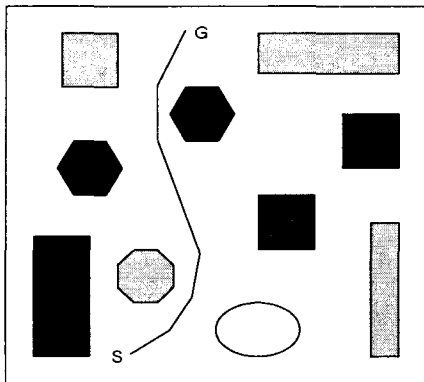


그림 15. 모의 실험-1
Fig. 15 Simulation-1

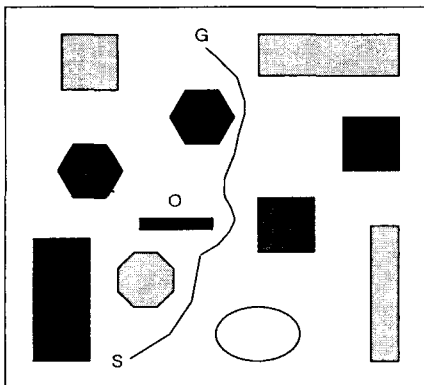


그림 16. 모의 실험-2
Fig. 16 Simulation-2.

그림 16은 주행 중에 환경 맵에 존재하지 않는 장애물 O를 센서 베이스 시스템이 감지한 후 경로를 수정하여 주행한 결과를 나타낸 것이다. 감지된 장애물 O를 회피하기 위하여 로컬 경로 계획기가 동작되어 방해물 회피 경로를 생성한다. 생성된 로컬 경로를 주행한 후 다시 이전의 전역 경로를 회복하여 목표점을 찾아간다

그림 18은 다양한 환경에서의 실험을 위하여 다른 환경에서 주행하는 것을 나타낸 것이다.

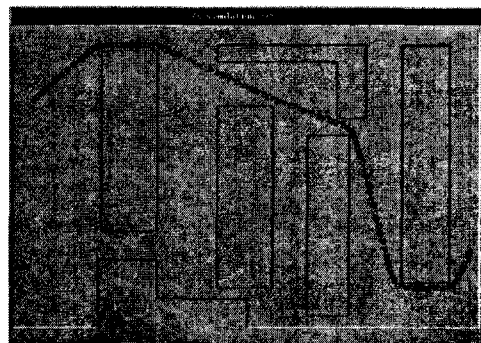


그림 17. 모의 실험-3
Fig. 17 Simulation-3

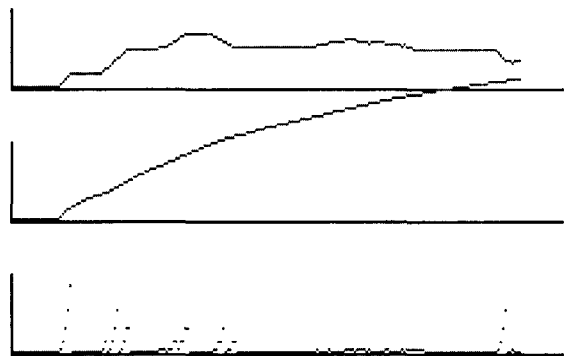


그림 18. 초기 단계의 실험 결과

- (a) 오차 함수 그래프
- (b) 누적 오차 함수 그래프
- (c) 오차의 변화량 함수 그래프

Fig. 18 The initial step's experiment result

- (a) Graph of error function
- (b) Cumulative graph of error function
- (c) Graph of Δe function

또한, 위와 같은 실험과 더불어, 본 논문에서 제안한 Fusion-SOFC 제어기의 객관적인 성능 평가를 위하여 그림 18, 19와 같은 그래프를 화면에 표시하고 적합도를 계산하였으며, 그림 20에 적합도의 변화를 나타내었다.

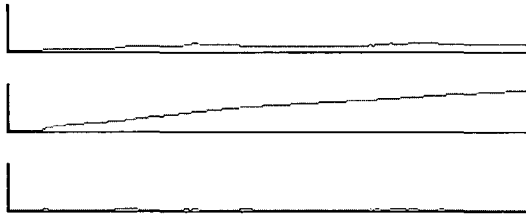


그림 19. 최종 실험 결과
Fig. 19 The final experiment result

그림 18은 최초 주행시의 $e_i, \sum_{i=1}^n e_i, \Delta e_i$ 값의 변화를 나타낸다. 그림 19는 최종 주행시의 변화량을 나타낸 것이다. 그림 18, 19에서 주행 성능을 평가하기 위하여 수평 위치 오차(lateral position error) e 를 누적하여 평가 함수의 입력 값으로 사용하였다.

- (a)는 매 순간 t 에서 발생하는 error의 크기
- (b)는 시간 t 까지 발생한 error의 합
- (c)는 시간 t 에서의 error의 변화량 Δe 를 나타내는데 경우에 따라서는 주행의 안정성을 나타내기도 한다.

초기의 주행 결과를 보면 굴곡이 있는 곳에서는 경로의 이탈 정도가 심하여 error의 크기도 크고 주행도 불안정한 형태를 이루고 있다. 그러나 Fusion-SOFC를 적용한 뒤의 최종 실험 결과는 경로의 이탈 정도가 적을 뿐만 아니라 error의 크기 역시 감소하였고 또한 주행 안정성도 뛰어난 것을 알 수 있다.

그림 20은 본 논문의 Fusion-SOFC 제어기의 적합도의 변화를 나타내고 있다. 초기의 적합도는 단순한 퍼지 제어를 수행했을 때와 같은 수치를 나타낸다. 이것은 본 논문의 Fusion-SOFC 제어기가 초기에는 일반 퍼지 시스템의 페러미터 값을 가지고 시작하기 때문이다.

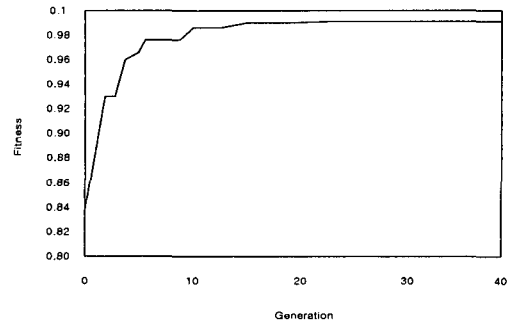


그림 20. 적합도 변화 그래프
Fig. 20 Fitness history graph

<표 3> 적합도의 비교
<Table 3> Comparison of Fitness

	Fuzzy control	F-SOFC	F-SOFC
Fitness	0.838	0.838	0.989

적합도 변화 그래프와 <표 3>의 적합도 비교에 서와 같이 기존의 퍼지 제어기에 비하여 성능이 약 18% 정도의 성능이 개선된 것을 알 수 있다.

V. 결 론

본 논문에서는 퍼지 제어기에 유전자 알고리즘과 진화 프로그램을 적용하고 또한, Fusion Method를 적용한 Fusion-자기 조직화 퍼지 제어 시스템(Fusion-SOFC)을 구성하였다. 본 논문에서 제안된 제어기의 성능 평가를 위하여 AGV 운행 제어에 적용하여 모의 실험을 하였다. 그 결과, Fusion Method를 사용하여 퍼지 추론 과정을 대체함으로써 기존의 퍼지 제어기에 비하여 훨씬 적은 수의 규칙을 이용하여 보다 간단하고 성능이 개선된 퍼지 제어를 구성할 수 있었다. 또한, COA 비퍼지화 방법을 AGV에 적용하였을 때, AGV 운행처럼 금지된 정보를 포함하고 있는 경우에는 COA 특성상 장애물을 피하지 못할 경우가 많아 적용이 어렵다. 이에 본 논문에서는 COA 방법을 변형하여 CLA 방법을 적용하여 이와 같은 문제점을 해결하였다. 또한, 본 논문의 Fusion-SOFC에는 퍼지 제어기의 단점으로

뽑히는 소속함수의 결과와 제어규칙의 결정에 GA와 EP를 적용하여 자기 조직화가 가능하게 함으로써 제어기 효율성을 18%정도 향상시켰으며, 아울러 객관성도 보장하였다. 앞으로는, 더 다양한 환경에서의 실험을 필요로 하며, 실제의 AGV를 구축하여 실험하는 것이 필요하고, 비전 시스템을 채택한 AGV의 제어기를 위한 연구 개발이 필요하다.

References

1. A Jain and G.Rodriguez, "Diagonalized Dynamic of Robot Manipulators, Proc. 1994 IEEE int. conf. on Robotics and Automation, IEEE Computer society press, pp.334~339, 1994.
2. Atsushi Konno, Masaru Uchiyama, "Modeling of a flexible Manipulator Dynamics Based upon Moizer's Model, "IROS '96, Japan, pp. 223~229, 1996.
3. B. McCane and T. Caelli, "Multi-scale adaptive segmentation using edge and region based attributes", Proc. KES '97, University of South Australia, pp. 72~80, 1997.
4. Georgess Girait, Mobile Robots, NATO ASI Series Vol. F11 Robotics and Artificial Intelligence, pp. 78~97, 1992.
5. J .J .Buckley and Y.Hayashi, Fuzzy neural networks : a survey, (Fuzzy Sets and Systems 66), pp. 1~13, 1994.
6. Jin Soo Kim, KuniakiSuzuki, "Force Control of a Flexible Manipulator Based on the Measurement of Link Deflections", IROS '96, Japan, pp. 238~245, 1996.
7. K. Le, "A proposed model of human consciousness system with applications if pattern recognition", Proc. KES '97, university of south Australia, pp. 159~167, 1997.
8. Karem Zita Haigh, Manuela M. veloso, "Interleaving Planning and Robot Execution for Asynchronous user Requests", IROS '96, Japan, pp. 148~155, 1996.
9. M. Caine, "The Design of shape Interactions Using Motion Constraints", Proc. 1994 IEEE int. conf. on Robotics and Automation, IEEE Computer Society Press, pp.336~371, 1994.
10. T. Fukuda, "Fuzzy-neuro-GA based intelligent robots", Computational Intelligence: Imitating Life, J.M.. Zurada, R.J. Marks II, C.J. Robinson, ED., IEEE Press, 252-263, 1994.
11. Yun-Bae Lee & young-Nam Na, "Applicatin of heuristic Control technique to path Planning of an autonomous guided Vehicle", INT. J. Computer Integrated Manufacturing, Vol. 7, No.2, pp. 123~128, 1994.
12. Yun-Bae Lee & Ki-dong Jang, "Curved-path and velocity Control of an autonomous guided vehicle using fuzzy logic", INT. J. Computer Integrated Manufacturing, Vol. 11, No.3, pp. 255~261, 1998.

이 영 신(Young-Sin Lee)

E-mail : yslee@www.seokang.ac.kr

1986년 광주대학교 전자계산학과 졸업(공학사)

1994년 조선대학교 대학원 전자계산학과 졸업(이학 석사)

1996년-현재 조선대학교 대학원 전자계산학과(박사 과정 재학중)

1986년-1996년 도로교통안전협회 전남지부 전산실 근무

1997년-현재 서강정보대학 생활정보과 전임강사

*관심분야 : 인공지능, 전문가시스템, 멀티미디어

나 영 남(Young-Nam Na)

E-mail : ynna@mail.chosun-c.ac.kr

1991년 조선대학교 전자계산학과 졸업(공학사)

1993년 조선대학교 대학원 전자계산학과 졸업(이학 석사)

1998년 조선대학교 대학원 전자계산학과 졸업(이학 박사)

1998년-현재 조선이공대학 정보통신과 전임강사

*관심분야 : 화상인식, 퍼지, 유전자알고리즘, CAI,

멀티미디어

이 윤 배(Yun-Bae Lee)

E-mail : yblee@mina.chosun.ac.kr

1980년 광운대학교 전자계산학과 졸업(공학사)

1993년 광운대학교 대학원 전자계산학과 졸업(이학 석사)

1998년 숭실대학교 대학원 전자계산학과 졸업(공학박사)

1988년-2000년 현재 조선대학교 컴퓨터공학부 교수

1999년-2000년 현재 광주광역시 시정정책자문위원회 위원

1996년-2000년 현재 광주광역시 및 전라남도 지역
정보화 추진 위원

2000년-2000년 현재 광주교육신문사 회장

1997년-2000년 조선대 정보과학대학장

1996년-1997년 호주 타스마니아대학 초빙교수

*관심분야 : 인공지능, 전문가시스템, 멀티미디어,
화상인식, 데이터 베이스