

유전 알고리즘을 이용한 퍼지 제어기 파라미터의 최적화

이승형* · 정성부** · 최용준* · 이승현* · 엄기환*

* 동국대학교 전자공학과 · ** 서일 대학 전자과

The Optimization of Fuzzy Controller Parameter using Genetic Algorithm

Seung-hyung Lee* · Sung-Boo Chung** · Yong-jun Choi* · Seung-hyun Lee* · Ki-whan Eom*

*Dongguk University · **Seoil College

E-mail : hyunga@cakra.dongguk.ac.kr

요 약

본 논문에서는 퍼지 논리 제어기에서 전문가의 지식없이 시행 착오법에 의해 최적화 되지 않은 제어 규칙을 이용하는 경우에도, 소속 함수 관계와 스케일링 팩터를 유전자 알고리즘으로 최적화하여 우수한 제어 성능을 갖는 지능 제어 방식을 제안한다. 제안하는 제어 방식은 실제 플랜트는 퍼지 논리를 이용해서 제어를 하되 먼저 오프 라인상에서 퍼지 제어기의 소속 함수 초기 변수값과 스케일링 팩터의 초기값을 유전 알고리즘으로 최적화시킨후 제어를 하는 직접 적용 제어 방식이다. 제안된 제어 방식의 유용성을 확인하기 위하여 비선형 시스템을 제어 대상으로 기존의 퍼지 제어 방식과 시뮬레이션을 통하여 비교 및 검토를 한다.

ABSTRACT

In this paper, we propose a method that optimizes the parameters of fuzzy logic controller : centers and widths of membership functions and scaling factors using genetic algorithm. Before fuzzy logic controller controls a plant in real time, first off it is optimized by genetic algorithm.

We select error and error variation between reference trajectory and real output for the input signals of fuzzy controller. We compared and investigated conventional fuzzy control method and proposed method through simulation and experiment using one link manipulator with nonlinear characteristic

1. 서 론

점점 더 발전되어 가는 자동 제어 시스템들은 복잡하고 불확실하며 비선형성이 강하여 고전적인 선형화 제어 방식으로는 적용의 한계에 부딪치곤 한다. 그래서 이런 문제점을 해결하기 위한 일환으로 최근에 각광을 받고 있는 인공 제어 방식은 사람의 자연적인 언어 체계를 표현하는데 효과적인 퍼지 시스템과 생물학적 신경 계통을 모방한 신경 회로망 그리고 자연 세계의 진화 현상에 기초한 계산 모델인 유전자 알고리즘 등을 이용하는 방식이 있다.[8] 이러한 인공 지능 시스템은 우리 인간의 지적인 기능들을 대신하여 처리해 줄 수 있는 시스템중에서 가장 적합한 모델이다.[6][8]

1965년 L.A.Zadeh 교수에 의해 제안된 퍼지 이론은 1974년 Mamdani의 모형 증기 기관의 제어 [12]에 성공적으로 적용함으로써 제어로의 적용

가능성을 제시한 이래로, 퍼지 제어 이론은 경험적 지식을 표현하는 강력한 도구로 등장하였다. 그러나 퍼지 제어 방식은 전문가로부터 제어 지식의 획득, 소속 함수 관계의 조정과 알맞은 스케일링 팩터(scaling factor) 선정 등의 문제점을 가지고 있어 원하는 제어 특성을 얻기 위해서는 여러번의 시행 착오를 거쳐야만 한다.[10][12]

John Holland에 의해 1975년에 개발된 유전자 알고리즘은 자연 선택의 원리와 자연계의 생물 유전학에 기본 이론을 둔 병렬적이고 전역적(global)인 탐색 알고리즘이다. 유전 알고리즘은 전역적인 최적의 해를 발견할 높은 가능성을 가진다는 장점과 목적 함수값(objective function value)에 대한 수학적 제약이 거의 없기 때문에 여러 분야에 적용할 수 있으며, 최근 들어 특히 신경 회로망과 퍼지 제어 방식과의 결합으로 그 응용 범위는 점점 늘어나고 있다.[6][7][8]

본 논문에서는 퍼지 논리 제어기에서 시행 착

오법에 의해 만들어진 최적화 되지 않은 제어 규칙을 이용하는 경우에도, 소속 함수 관계와 스케일링 팩터를 유전자 알고리즘으로 최적화하여 우수한 제어 성능을 갖게 하는 지능 제어 방식을 제안한다. 제안하는 제어 방식은 실제 플랜트는 퍼지 논리를 이용해서 제어를 하되 먼저 오프라인상에서 퍼지 제어기의 소속 함수 초기 변수값과 스케일링 팩터의 초기값을 유전 알고리즘으로 최적화시킨후 제어를 하는 직접 적용 제어 방식이다. 제안된 제어 방식의 유용성을 확인하기 위하여 비선형 시스템인 단일 링크 매니플레이터를 제어 대상으로 기존의 퍼지 제어 방식과 시뮬레이션을 통하여 비교 및 검토를 한다.

II. 퍼지 논리 제어 방식

퍼지 제어기는 시스템 제어 분야의 현장 경험이 많은 전문가의 지식을 언어적 형태로 기술한 제어 규칙을 작성하여 시스템의 출력이 원하는 기준 입력을 추종하도록 제어 입력을 계산한다.

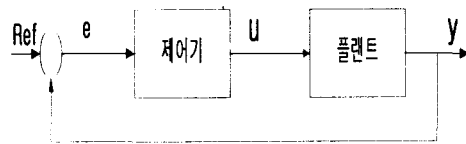


그림 2.1 제어 시스템의 일반적인 형태

제어 시스템의 일반적인 형태는 그림 2.1과 같으며 제어기 부분을 그림 2.2와 같은 퍼지 제어기로 대체하면 퍼지 제어 시스템의 일반적인 형태가 된다.[1][2][3][4]

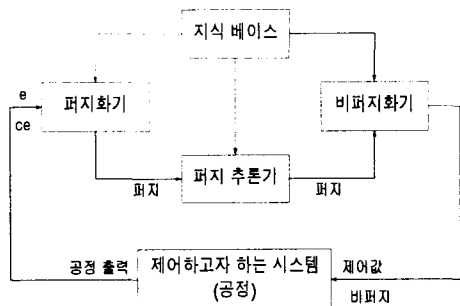


그림 2.2 퍼지 논리 제어기의 기본 구성도

퍼지 제어기의 역할은 오차(e)를 입력받아서 제어 입력(u)를 계산한다. 기존 제어기에서는 일·출력이 모두 명확한 값으로 주어진다. 그러나 퍼지 제어기는 제어 규칙을 언어적으로 기술하여 제어를 수행하기 때문에 비선형 제어 시스템에도

적용성이 우월하다. 여기서 e는 오차, ce는 오차의 변화분, u는 제어입력을 나타낸다. 퍼지 제어기는 모호한 오차 입력으로부터 출력을 얻어내야 하는데 그림 2.2에서와 같이 퍼지화기, 규칙 베이스, 퍼지 추론기, 비퍼지화기로 구성된다.

퍼지화기는 제어기 입력 변수의 값을 측정하고, 입력 변수값의 영역을 이에 상응하는 전체 집합으로 크기 변환(scale mapping)시킨다. 또 입력값의 퍼지화를 수행한다. 즉, 입력 데이터를 적절한 언어적인 값으로 변환시킨다. 여기서 언어값은 일반적으로 입력부 전체 집합내에서 정의된 퍼지 집합들의 라벨을 사용한다.

지식 베이스에는 제어 대상부에 대한 지식과 제어 목적이 기술된다. 지식 베이스는 구체적으로 데이터 베이스와 언어적(퍼지) 제어 규칙부로 구성된다. 데이터 베이스는 퍼지 논리 제어에서의 퍼지 데이터를 조작하고, 언어적 제어 규칙을 정의하는데 필요한 사항들을 정의하고 있으며, 제어 규칙부는 제어 목적과 전문가의 제어 방침을 언어적 규칙들로 나타낸다.

퍼지 추론기는 퍼지 논리 제어기의 실행부이다. 이는 퍼지 관계와 퍼지 논리의 추론 규칙을 도입하여 인간의 의사 결정 방식을 묘사하여 퍼지 제어 입력을 구해주는 기능을 한다.

비퍼지화기는 출력 퍼지값을 실제 제어 입력의 전체 집합에 맞추어 크기 변환시키며, 출력(제어 입력)의 비퍼지화를 수행한다. 즉, 위에서 계산된 퍼지값은 실제 제어 입력으로 사용할 수 없으므로 이를 명확한 비퍼지값으로 변환시킨다. 비퍼지화의 방법에는 여러 가지가 있으나 주로 무게 중심법이 쓰인다.

III. 유전 알고리즘

John Holland에 의해 개발된 유전 알고리즘의 연구 목적은 두 가지로, 첫 번째는 자연계 시스템의 적응 과정을 합축 강건(robustness)하게 설명하기 위한 것이고, 두 번째는 자연계 시스템의 골격(mechanism)을 그대로 유지한 채 인공적인 시스템 모델로 만들려는 것이다.

유전 알고리즘은 미지의 함수 $Y = f(x)$ 를 최적화하는 해 x를 찾는 탐색 알고리즘이다. 전역적인 최적화 해를 구하기 위하여 재생산, 교배, 돌연변이라는 유전 연산자를 사용한다.

재생산은 각각의 스트링이 가지는 적합도에 따라 그 스트링을 복제하는 과정이며 이 때 적합도 함수는 사용자가 최대화하거나 최소화하기를 원하는 어떠한 형태의 함수도 가능하다. 이 과정은 적합도가 높은 개체일수록 다음 세대에 더 많은 자손을 가질 확률이 높음을 의미하며 이는 주어진 환경에 더 잘 적응하는 개체만이 살아남는다는 자연 선택의 원리를 담고 있다.

위의 재생산 과정은 탐색 방향을 적합도가 가장 높은 개체쪽으로 향하게 되지만 새로운 개체를

생산하지는 못한다. 자연계에서 자손은 부모 세대로부터 유전자를 이어받으면서도 부모와는 전혀 다른 형질을 가지듯이 유전 알고리즘에서도 이 과정을 모방한 것이 교배 과정이다. 두 문자열을 교배시키는 과정은 그림 3.1과 같다.

이는 교배율(crossover rate : Pc)을 가지는 확률적 과정임에도 불구하고 재생산과 함께 쓰여 개체간의 정보를 교환하고 스트링의 좋은 부분을 결합하는 효과적인 수단이 된다. 즉 적합도가 높은 개체들을 빠르게 결합하여 현재 집단의 전역에 확산시키는 역할을 하며 다른 최적화 알고리즘에 없는 유전 알고리즘에서 가장 중요하고 독특한 과정이다.

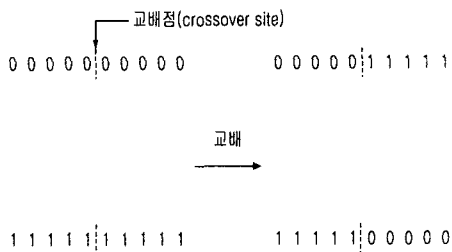


그림 3.1 교배 과정

재생산과 교배 과정은 집단의 개체들이 현재 가지고 있는 정보들을 이용해서 결합하고 탐색하는 과정으로 현재 집단에 존재하지 않는 새로운 정보를 제공해 준다. 이는 아주 낮은 돌연변이율 (mutation rate : Pm, 보통 0.05 이하의 값을 사용)로 스트링의 한 비트를 0에서 1로 혹은 1에서 0으로 우연히 바꿈으로써 구현된다. 이 과정은 과거에 잃어버렸던 중요한 정보를 되찾을 수도 있으며 탐색의 방향이 지역적 극값으로 변할 경우 여기서 벗어나도록 도움을 준다. 어느 한 문자열에서 돌연변이가 일어나는 과정은 그림 3.2와 같다.

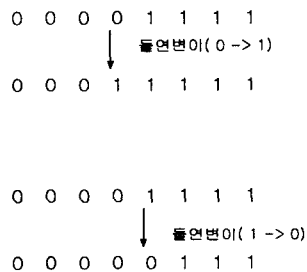


그림 3.2 돌연변이 과정

이러한 유전 연산자를 적용하기 위해서는 풀고자 하는 문제의 변수값을 이진 스트링으로 표현

해야 한다. 개체의 코딩 방법은 문제의 변수가 이진치이거나 다른 이산치를 갖는 경우에 특히 자연스러운 표현법이다. 만약 문제가 실수의 값을 가질 경우에는 이진화된 정수 형태로 변환하여 사용한다.

유전 알고리즘은 10진수로 표현된 파라미터를 일정한 길이의 문자열로 변환시켜 사용하는데 주로 무부호 2진 스트링을 사용한다. 이런 개체들을 모아 하나의 집단을 구성하여 진화를 시키므로 한 점씩 탐색해 나가는 고전적인 알고리즘과는 달리 많은 점을 동시에 탐색할 수 있는 능력을 가지게 된다.

목적 함수(objective function) 즉, 최적화(최대화 또는 최소화)하고자 하는 함수는 각 개체의 적합도를 평가하는 기반이다. 그러나 목적함수의 값의 범위는 문제마다 다르기 때문에 보통 정해진 구간 사이의 양수의 값을 갖도록 표준화된 값을 적합도로 사용한다. 엄격히 구별하자면 표준화되기 전의 적합도의 값을 로 적합도(raw fitness)라고 하면 표준화되어서 실제로 선택의 기준이 되는 함수를 적합 함수라고 한다.

이러한 사항들을 고려한 유전 알고리즘은 모델에 따라 약간의 차이는 있으나 전체적인 알고리즘의 흐름은 다음과 같이 나타낸다.[1][6][7][8]

- ① M개의 개체를 가진 최초의 개체군 P(0)를 형성한다. (세대 t = 0)
- ② 개체군 P(0)에 있는 모든 개체에 대해 적합도를 평가한다.
- ③ 원하는 수준의 해를 가진 개체가 발견되었으면 수행을 멈춘다.
- ④ P(t)로부터 적합도에 기반해 부모 개체를 선택하고 이를 교차나 돌연변이와 같은 유전 연산자에 의해 변형함으로써 새로운 M개의 개체를 생성한다.
- ⑤ 새로운 개체군을 P(t+1)라 하고 세대의 수를 하나 증가한다. (t = t+1)
- ⑥ 앞의 2단계로 가서 위의 과정을 반복한다.

IV. 제안한 제어 방식

일반 퍼지 제어 방식에서는 소속 함수의 변수값과 비퍼지화의 스케일링 팩터를 시행 착오에 의하여 정하게 되므로, 최적화 된 값은 아니다. 그러므로 제안하는 제어 방식에서는 이러한 값들을 최적화하여 우수한 제어 성능을 얻기 위하여 유전 알고리즘을 이용한다. 그러나 유전 알고리즘은 계산량이 상당히 많기 때문에 간단한 하드웨어에서 실시간으로 적용하기에는 무리가 따른다. 그래서 시뮬레이션에서 유전 알고리즘을 이용하여 최적의 소속 함수의 변수값과 비퍼지화 스케일링 팩터를 구하고, 이렇게 구한 값들을 초기값으로 하여 퍼지 제어를 구성하고 실제 하드웨어 상에서 실시간으로 실험을 한다. 그림 4.1은 제안한 제어 방식의 블록선도이다

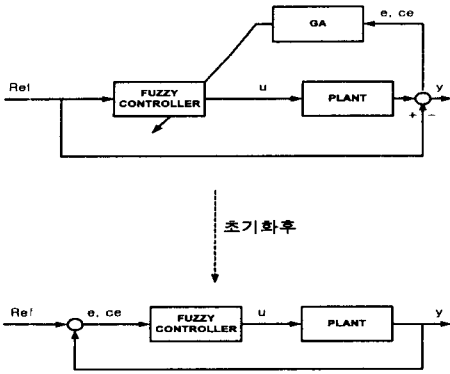


그림 4.1 제안한 퍼지 제어 방식의 제어 방식

그림 4.1에서 Ref는 원하는 신호, y는 실제 출력, e와 ce는 각각 오차와 오차의 변화율이며, u는 퍼지 제어기에서 계산된 플랜트에 가하는 제어 입력이다.

그림 4.2는 본 논문의 제어 규칙에서 사용되는 소속 함수이다. 이 가우시안 형태의 함수는 비교적 비선형성에 적응성이 뛰어나며, 이 함수의 중심 c_i 와 표준 편차 σ_i 를 유전 알고리즘으로 최적화 시킨다. 소속 함수식은 식 (4.1)와 같다.

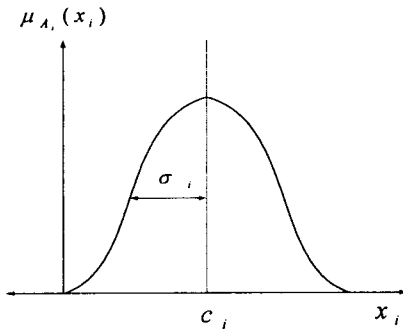


그림 4.2 가우시안 형태의 소속 함수

$$\mu_{A_i}(x_i) = \exp\left[-\frac{1}{2}\left(\frac{x_i - c_i}{\sigma_i}\right)^2\right], \quad (-\infty < x_i < \infty, \sigma_i > 0) \quad (4.1)$$

여기서, x_i 는 입력 변수의 값이고, c_i 는 가우시안 함수의 중심이고, σ_i 는 표준 편차에 해당하며, 마지막으로 $\mu_{A_i}(x_i)$ 는 소속 함수에 대한 입력변수의 소속 정도를 나타낸다.

본 논문에서 사용하는 제어 규칙은 일반적인 서보 시스템의 위치 및 속도 시뮬레이션에 의해 시

행 착오적으로 구한 것으로 표 4.1과 같으며, 이를 이용하여 퍼지 제어를 한다.

표 4.1 퍼지 제어 규칙

e \ ce	NL	NM	NS	ZE	PS	PM	PL
NL				NL	PL	PL	PL
NM				NM	PM	PM	PL
NS			PS	NS	PS	PM	PL
ZE	NL	NM	NS	ZE	PS	PM	PL
PS			NS	PS	NS		
PM				PM			
PL				PL			

퍼지 추론 방식은 Mamdani의 Max-Min법을 사용하며, 비퍼지화의 방법은 일반적으로 가장 많이 쓰이는 무게 중심법을 이용한다.

퍼지 논리 제어기의 변수를 최적화하기 위하여 소속 함수의 중심값과 표준 편차 그리고 스케일 팩터에 유전 알고리즘을 이용한다.

최적화를 시도하고자 사용하는 유전 알고리즘의 적합 함수는 식 (4.2)과 같으며, 값이 작을수록 적합도가 높다.

Fitness Function :

$$F = \sum_{i=1}^n (y_d(i) - y(i))^2 = \sum_{i=1}^n (e_i)^2 \quad (4.2)$$

여기서 y_d 는 원하는 기준 궤적이 되며, y 는 실제 출력이다.

V. 시뮬레이션

제안한 제어 방식의 유용성을 입증하기 위하여 식 (5.1)로 표현되는 단일 링크 매니플레이터 [9][11]에 대하여 IBM 호환 PC Pentium II 400MHz 상에서 MatLab 언어를 사용하여 시뮬레이션 한다.[5]

$$m l^2 \ddot{\theta} + B \dot{\theta} + m g l \cos(\theta) = \tau \quad (5.1)$$

여기서 θ , $\dot{\theta}$, $\ddot{\theta}$ 는 각각 단일 링크 매니플레이터의 각변위, 각속도, 각가속도를 나타내고, τ 는 매니플레이터의 입력으로써 관절에 가해지는 토크이며, m , l , B , g 는 각각 단일 링크 매니플레이터의 질량, 길이, 마찰 계수, 중력 가속도를 나타내며, 설정치는 각각 $m = 0.5[\text{kg}]$, $l = 0.18[\text{m}]$,

$B = 0.4 \text{ [kg} \cdot \text{m}^2/\text{s]}, g = 9.8 \text{ [m/s}^2\text{]}$ 이다. 식(5.1)의 시스템에서 $(\theta \ \dot{\theta})^T = (x_1 \ x_2)^T$ 로 정의하면 식 (5.2)의 비선형 상태 공간 모델로 된다.

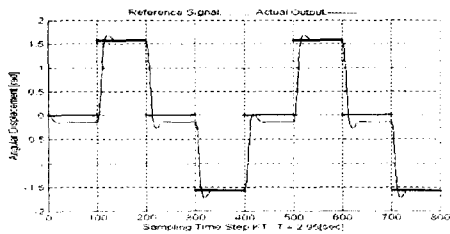
$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{g}{l} \cos(x_1) - \frac{B}{ml^2} x_2 + \frac{1}{ml^2} \tau \end{aligned} \quad (5.2)$$

식 (5.2)에 대하여 Runge-Kutta fourth order 알고리즘을 이용하고 2.95[ms]를 샘플링 시간 스텝으로 하여 비선형시스템의 동정과 제어 방식에 대하여 기존의 퍼지 제어기를 이용한 직접 제어 방식과 제안한 방식을 시뮬레이션 하여 제어 성능을 비교 검토한다.

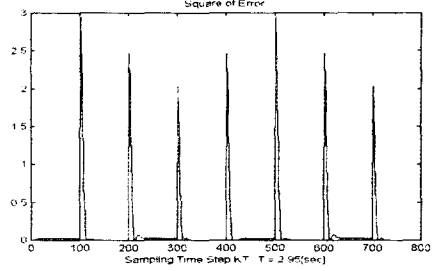
퍼지 제어기는 표 4.1의 퍼지 제어 규칙을 가지고 적용한다. 소속 함수는 가우시안 함수를 사용하였으며, 초기 중심값과 표준편차는 [-1 1]사이의 간격을 소속 함수의 수만큼 등간격으로 잡아주었다. 오차와 오차의 변화량에 대한 스케일링 팩터는 동일하게 1/30로 주었으며, 출력의 스케일링 팩터는 1000으로 하였다. 유전 알고리즘에 적용하는 제어 파라미터인 교배율(Pc)과 돌연변이율(Pm)은 온-라인 및 오프-라인 성능이 우수한 Dejong의 값[1]을 이용하게 되는데, Pc = 0.6, Pm = 0.001을 사용하게 된다. 개체군의 크기는 45개의 최적 변수를 8 비트씩 하여 총 350bit의 200개 개체, 종료 조건인 세대수는 30 세대까지 한다. 기준 궤적은 식 (5.3)으로 주어지며, 오차 비교를 위해서는 식(5.4)를 사용한다. 단일 링크 매이플레이터의 초기 조건은 각변위 0, 각속도 0로 하였으며, 시뮬레이션 결과는 그림 5.1, 5.2와 같다.

$$y_d = \begin{cases} 0 & 1 \leq i \leq 100, \quad 400 < i \leq 500, \\ \pi/2 & 200 < i \leq 300, \quad 600 < i \leq 700 \\ -\pi/2 & 100 < i \leq 200, \quad 500 < i \leq 600 \\ -\pi/2 & 300 < i \leq 400, \quad 700 < i \leq 800 \end{cases} \quad (5.3)$$

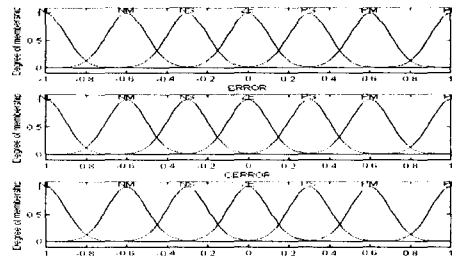
$$SSE = \sum_{i=1}^n (e_i)^2, \quad (i = 1 \dots 800) \quad (5.4)$$



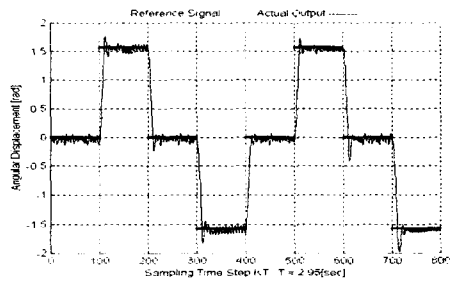
(a) 각변위 추종 곡선



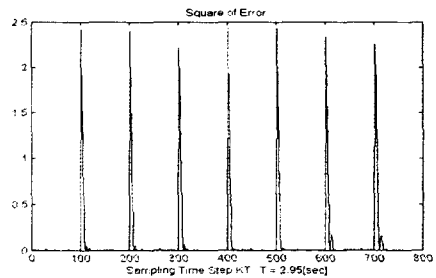
(b) 제곱 오차 곡선



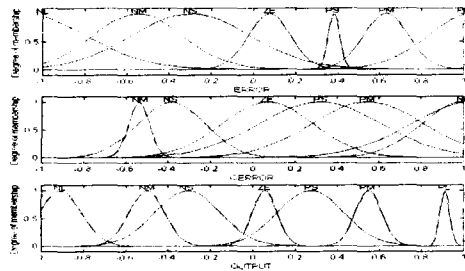
(c) 입력 및 출력의 소속 함수
그림 5.1 기존 퍼지 제어방식의 특성



(a) 각변위 추종 곡선



(b) 제곱 오차 곡선



(c) 입력 및 출력의 소속 함수

그림 5.2 제안한 퍼지 제어 방식의 특성

그림 5.1과 그림 5.2에서 확인하듯이 제안한 제어 방식이 추종 특성이나 제곱 오차면에서 훨씬 더 양호함을 알 수 있다. 이상과 같이 기존의 퍼지 제어 방식과 본 논문에서 제안한 방식을 비교·검토한 결과는 표 5.1과 같다.

표 5.1 제어 방식의 비교

제어 방식		기	제
종류		존	안
		제어	한
		제어	제어
		방	방
		식	식
제어 규칙 개수		25	25
스케일링 팩 터	GE	0.0333	0.8902
	GC	0.0333	0.6157
	GU	1000	2117.6
유 전 알고리즘 적 용 파라메터	표준편차	미 적 용	21
	중심값		21
	스케일링 팩 터		3
	총 개수		45(350Bit)
제곱 오차의 합(SSE)		101.4589	75.2072

VI. 결 론

본 논문에서는 퍼지 논리 제어 방식에서 전문가의 지식없이 시행 착오법에 의한 최적화 되지 않은 제어 규칙을 이용하는 경우에, 소속 함수 관계와 스케일링 팩터를 유전자 알고리즘으로 최적화하여 우수한 제어 성능을 갖는 지능 제어 방식을 제안하였다. 제안한 방식은 유전 알고리즘을 이용하여 퍼지 소속 함수의 중심값과 표준 편차 그리고 오차와 오차의 변화량에 대한 스케일링 팩터를 최적화한 값으로 구하고, 이 값들을 퍼지 제어기의 초기값으로 하여 실제 하드웨어에 실시간으로 적용을 하는 방식이다.

유전 알고리즘으로 최적화된 파라미터를 퍼지 제어기에 적용시켜, 단일 링크 매니퓰레이터를 이

용하여 추종제어에 대한 시뮬레이션을 하여 기존 퍼지 제어 방식과 비교 검토한 결과, 제안한 제어 방식이 기존의 퍼지 제어 방식보다 추종 특성이나 제곱 오차 합 성능이 향상됨을 확인할 수 있었다.

참고문헌

- [1] 임영도, 이상부, " 퍼지·신경망·유전 진화", 도서출판 영과일, 1997.
- [2] Li-Xin Wang, "A Course in Fuzzy Systems and Control", Prentice Hall. Inc., 1997.
- [3] Li-Xin Wang, "Adaptive Fuzzy Systems and Control(Design and Stability Analysis)", Prentice-Hall. Inc., 1994.
- [4] D.Driankov, H.Hellendoorn, M.Reifrank, "An Introduction to Fuzzy Control", Springer, 1996.
- [5] J.Wesley Hines, "Fuzzy and Neural Approaches in Engineering", John Wiley & Sons, 1997.
- [6] Mitsuo Gen, Runwei Cheng, "Genetic Algorithms & Engineering Design", John Wiley & Sons. Inc., 1997.
- [7] G.Winter, J.Periaux, M.Galán, P.Cuesta, "Genetic Algorithms in Engineering and Computer Science", John Wiley & Sons., 1995.
- [8] Chin-Teng Lin and C.S.George Lee, "Neural Fuzzy Systems", Prentice Hall PTR, 1996.
- [9] 이용구, "퍼지-신경 회로망과 신경 회로망을 이용한 비선형 시스템의 혼합 동정과 제어 방식", 동국대학교 대학원 박사 논문, 1996.
- [10] Ming Kan Gu, et al., "A Cascade Architecture of Adaptive Fuzzy Controllers for Inverted Pendulums", FUZZ·IEEE, Vol.2, pp. 1514-1519, 1994.
- [11] M.Yuan, G.S.Hong and A.N.Poo, "Neural Adaptive Controller : Application to Robot Manipulator", The Third International Conference on Automation, Robotics and Computer Vision, pp. 1734-1737, 1994.
- [12] H.Ying, "Practical Design of Nonlinear Fuzzy Controllers with Stability Analysis for Regulating Processes with Unknown Mathematical Models", Automatica, Vol.30, No.7, pp. 1185-1195, 1994.