

유전자 알고리즘을 이용한 비선형 시스템의 퍼지-신경 회로망 모델링

이승형^{*} · 최용준^{*} · 김주용^{*} · 김한웅^{**} · 김경수^{*} · 엄기환^{*}
^{*}동국대학교 전자공학과 · ^{**}안성산업대 전자공학과

Fuzzy-Neural Network Modeling of Nonlinear Systems using Genetic Algorithms

Seung-hyung Lee^{*} · Yong-joon Choi^{*} · Joo-Woong Kim^{*} · Han-woong Kim^{**} · Kyung-soo Kim^{*} · Ki-Whan Um^{*}

^{*}Dongguk University · ^{**}Ansung National University

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

요 약

본 논문에서는 유전자 알고리즘을 이용하여 불확실한 비선형 시스템의 퍼지-신경 회로망 모델링을 제안하였다. 제안한 퍼지-신경 회로망 모델링을 위한 학습 알고리즘은 다음과 같은 세 단계로 나누어 진행된다. 첫 번째 단계에서는 퍼지 모델의 소속 함수의 중심값과 표준편차를 구하여 초기 퍼지 소속 함수를 결정한다. 두 번째 단계에서는 새로운 알고리즘을 통하여 언어적 퍼지 규칙을 만든다. 마지막 세 번째 단계에서는 유전자 알고리즘을 이용하여 중심값과 표준편차를 최적화함으로써 퍼지 모델의 소속 함수를 조절한다. 제안된 유전자 알고리즘의 장점은 흔히 신경 회로망에서 널리 쓰이는 역전파 알고리즘이 갖는 지역 최소점에 빠지는 현상이 없다는 것이다. 제안한 알고리즘의 유용성을 확인하기 위하여 일반적으로 가장 많이 쓰이는 비선형 시스템에 대하여 시뮬레이션 하여 확인하였다.

1. 서 론

최근 로봇 등과 같이 복잡하고 불확실하며 비선형성이 강한 기구계를 가진 역학이나 이러한 것을 포함한 복합 시스템의 제어가 급증하고 있다. 이것은 현대의 과학 기술, 공업 기술의 급속한 진보와 깊은 연관성이 있으며, 이것에 따라 제어 이론 및 기술도 급속히 고도화되고 복잡화 되어 가고 있다.[1][2][3][4]

현대 제어 이론의 성숙과 함께 1970년대에는 최적 제어에 대한 연구가 매우 활발하였으며, 80년대 후반에는 사람의 자연적인 언어 체계를 표현하는데에 효과적인 퍼지 이론과 생물학적 신경 계통을 모방한 신경 회로망 이론과 자연 세계의 진화 현상에 기초한 계산 모델로써 최적화에 뛰어난 성능을 보여 주는 유전자 알고리즘이 제어 공학자들에 의해 많은 연구의 대상이 되었다.[2][6][7][8]

신경 회로망은 학습 기능과 대량 병렬성을 보유하고 자기 조정 기능을 갖는 유연성이 뛰어난 계산 모델이지만, 지식 표현이 구조적으로 분명하지 않고 전문가의 애매한 지식을 반영하기가 힘들며 학습하는데 시간이 오래 걸리는 단점이 있

다. 퍼지 시스템은 지식의 논리적, 구조적 표현을 명확히 할 수 있고, 퍼지 추론 등의 퍼지 연산을 통하여 퍼지 정보 처리를 할 수 있고, 복잡한 시스템에 대한 개략적 기술을 가능하게 하고 자연적으로 표현을 사용한 고위의 정보 처리가 가능한 시스템이다. 그러나 학습 능력이 없고 소속 함수와 규칙에 대한 조정 능력이 없어 변화하는 환경에 대한 적응성이 떨어지고 계산 시간이 많이 걸리는 단점이 있다. 그래서 최근에는 신경망과 퍼지 시스템을 결합하여 상호 보완하는 연구가 활발히 진행되고 있다.[2][3][4][6] 또한 유전자 알고리즘을 이용함으로써 퍼지-신경 회로망 단일로 쓰일 때보다 어느 정도의 수준에서 목표치에 만족하는 정확도를 얻어내면서 빠른 학습 시간을 얻을 수 있다.

본 논문에서는 유전자 알고리즘을 이용하여 불확실한 비선형 시스템의 퍼지-신경 회로망 모델링을 제안한다. 제안하는 방식은 기존의 신경망과 퍼지 시스템의 상호 보완을 위한 퍼지-신경 회로망에 최적화에 대해서 가장 우수한 성능을 보여주는 유전자 알고리즘을 추가, 합성한다. 퍼지-신경 회로망의 학습 알고리즘은 세 단계로 나누어

서 진행하게 된다.

첫 번째 단계에서는 퍼지 모델의 소속 함수의 중심값과 표준편차를 구하여 초기 퍼지 소속 함수를 결정한다. 두 번째 단계에서는 새로운 알고리즘을 통하여 언어적 퍼지 규칙을 만든다. 마지막 세 번째 단계에서는 유전자 알고리즘을 이용하여 중심값과 표준편차를 최적화 함으로써 퍼지 모델의 소속 함수를 조절한다.

제안한 알고리즘의 유용성을 확인하기 위하여 제어 대상으로서 일반적으로 가장 많이 쓰이는 비선형 시스템 중 입력 신호와 관계되는 변수들이 시불변 특성을 갖는 시스템인 단일 링크 매니플레이터에 대하여 시뮬레이션 한다.

II. 유전자 알고리즘

인공 유전 시스템인 유전자 알고리즘(이하 GA)은 자연 세계의 진화 현상에 기초한 계산 모델로써 1975년 미시건 대학교의 John Holland와 그의 동료와 제자들에 의해 개발되어 왔다.[7][8]

이 GA는 자연 도태의 유전적인 메카니즘에 기초한 탐색 알고리즘으로써 세대를 거듭함에 따라 어떤 최적화 하는 해에 수렴 또는 도달하는 탐색 알고리즘이다.

자연계의 모든 생물들은 유전인자인 염색체에 의해서 생식과 유전 등 진화과정을 반복하여 발전해 나간다. GA에서는 생물학적 유전인자인 염색체에 해당하는 문자열을 가지고 생물과 같은 재생산(reproduction), 교배(crossover), 돌연변이(mutation)를 거쳐서 다음 세대의 새로운 자손(offspring)을 인공적으로 만들어낸다. 자연의 생물 유전을 모방한 연산자들을 반복적으로 적용하여 적합한 해를 탐색한다.

GA는 기존의 다른 알고리즘보다 많은 최적화 문제에서 전역적이고 견실한 최적화 방법이다. 고전적인 제어기 알고리즘들은 제어기에 대한 충분한 지식과 시스템의 수학적 해석 및 계산에 의해서 설계되었다. 이렇게 설계된 시스템도 주어진 환경하에서 지역적일 수도 있었다. 그러나 GA는 전역적인 최적의 해를 발견할 가능성이 매우 높다는 장점과 목적 함수값에 대한 수학적 제약이 거의 없기 때문에 여러 분야에 적용 시킬수 있다.

III. 퍼지-신경 회로망

퍼지-신경 회로망(이하 NF)은 그림 1과 같이 다층 신경 회로망으로 총 5개의 층으로 이루어졌다. 본 논문에서는 2개의 입력(오차와 오차의 변화분)과 1개의 출력을 갖는 모델을 구성한다. 그래서 첫 번째 층에는 2개의 노드가 있고, 다섯 번째 층에는 1개의 노드가 있다. 층 1의 노드는 입력 노드이고, 다음 층에 곧바로 입력 신호를 전송한다. 층 5는 출력층이 된다. 층 2와 4의 노드들은 입력 및 출력 퍼지 언어 변수들을 표현하는데 소속 함수 같은 역할을 하는 'term 노드'이다. 종 형태의 함수를 소속 함수를 표현하는데 선택하였는데, 중심값 m 과 표준편차 σ 는 학습 과정을 통하여 조정하여 나간다. 첫 번째와 두 번째 입력 변수 2개의 퍼지 집합은 각 n_1 과 n_2 의 언어적 항목으로 구성이 된다. positive large(PL), positive medium(PM), positive small(PS), zero(ZE), negative small(NS), negative medium(NM), negative large(NL)과 같은 언어적 항목은 term 노드에서 내림차순으로 번호가 주어진다. 그러므로 n_1+n_2 노드와 n_3 노드는 각각 층 2와 4에 포함되는데, 그것은 입력 및 출력의 언어적 변수들을 지시하기 위해서이다.

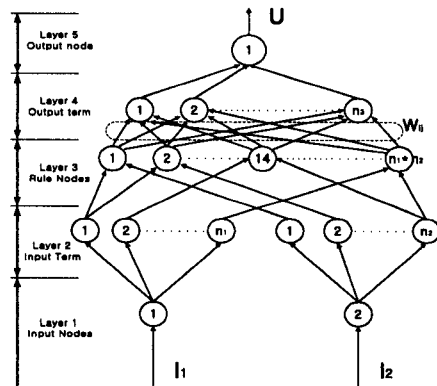


그림 1. 뉴로 퍼지 모델의 구성

Fig 1. Construction of the Fuzzy-Neural Network model

층 3에서 각 노드는 '규칙 노드'이고 하나의 퍼지 규칙을 나타낸다. 전체적으로 층 3에는 $n_1 \times n_2$ 의 노드가 있고, 2개의 언어적 변수에 대한 퍼지

규칙 근간을 형성한다. 층 3과 층 4의 연결은 각각 규칙 노드의 전체 조건과 결과를 정의한다. 각 규칙 노드에 대하여, 입력 term노드로부터 2개의 고정된 연결이 있다. 층 4의 연결에 있어서는 점선으로 둘러싸인 부분, 제어 상황을 변경함으로써 응답을 조절할 수 있다. 이와는 반대로, 층 2와 5의 연결은 입력 및 출력 노드와 그들의 대응되는 term 노드 사이에 고정된 상태로 있다. NF 모델은 층 2와 4에서 각 노드에 대한 가우시안 형태의 소속 함수를 나타내는 변수와 층 4의 연결을 변경함으로써 규칙과 그들의 소속 함수를 조절시킬 수 있다.

IV. 제안한 학습 알고리즘

불확실한 비선형 시스템의 NF 모델링을 위한 학습 알고리즘은 다음과 같은 세 단계로 나누어 진행된다.

1. 학습 1단계

소속 함수의 초기 중심값과 표준편차는 통계적 클러스팅에 연속적인 자기 조직화 학습 방법에 의하여 결정되어진다. 이 방법은 데이터가 존재하는 입력 및 출력 공간의 영역에서만 분포하는 소속 함수의 영역을 위치시킴으로써 효과적으로 퍼지 신경망의 자원을 할당할 수 있도록 한다. kononen의 self-organized feature-Map (SOM) 알고리즘은 소속 함수의 중심 m_i 를 알아내는데 사용한다.

$$\|x(t) - m_{closest}(t)\| = \min_{1 \leq i \leq k} \{ \|x(t) - m_i(t)\| \} \quad (1)$$

$$m_{closest}(t+1) = m_{closest}(t) + \alpha(t) [x(t) - m_{closest}(t)] \quad (2)$$

$$m_i(t+1) = m_i(t) \text{ for } m_i \neq m_{closest} \quad (3)$$

여기서 $\alpha(t)$ 는 단조적으로 감소하는 스칼라 학습율이고 k 는 퍼지 분할의 수이다. 여기서 적용한 공식은 각 입력과 출력의 언어 변수에 대하여 독립적으로 실행된다.

일단 소속 함수의 평균을 알게 되면, 그 소속 함수들의 표준편차들은 식 (4)를 최소화 시킴으로써 $N_{nearest-neighbors}$ heuristic에 의하여 결정 가능하

게 된다.

$$E = \frac{1}{2} \sum_{i=1}^N \left\{ \sum_{j \in N_{nearest}} \left(\frac{m_i - m_j}{\sigma_i} \right)^2 - r \right\}^2 \quad (4)$$

여기서 r 은 보통 1.0에서 2.0 범위의 접치는 변수이다. 우리는 세 번째 단계에서 소속 함수의 중심과 표준편차를 최적으로 조절할 것이기 때문에, 표준편차는 이번 단계에서는 식 (5)와 같이 간단히 첫 번째 nearest-neighbors heuristic에 의해서 결정될 수 있다.

$$\sigma_i = \frac{|m_i - m_j|}{r} \quad (5)$$

2. 학습 2단계

소속 함수의 변수들을 알아낸 후에는, 각 규칙 노드의 파이어링 강도(firing strength)를 얻을 수 있다. 이들 rule-firing 강도(O_{kt}^3)와 층 4에서의 term 노드의 출력(O_{kt}^4)을 이용하여, $n_1 \times n_2$ 규칙을 알아내기 위한 각 규칙 노드에 대한 올바른 연결 강도를 결정한다.

학습 2단계 알고리즘의 절차는 다음과 같다.

Step 1 : 층 3의 각 규칙 노드에 대하여 n_3 매칭 요소를 구축한다. 이 경우에는 $n_1 \times n_2 \times n_3$ 개의 매칭 요소를 가진다. 각 매칭 요소는 M_{ij} 로 표시되고, 여기서 첨자 i ($= 1, 2, 3, \dots, n_1 \times n_2$)는 규칙 노드 지수이고 첨자 j ($= 1, 2, \dots, n_3$)는 출력 항목 노드 지수(output-term-node-index)이다.

Step 2 : M_{ij} 는 다음과 같은 의사코드에 따라 계산되어진다.

```

For i = 1, 2, . . . ,  $n_1 \times n_2$ 
  For j = 1, 2, . . . ,  $n_3$ 
    For k = 1, 2, . . . , N
       $M_{ij} = \begin{cases} M_{ij} + O_k^3, & O_k^3 \text{가 집합 } O_k^3 \text{에서 최대 요소일 경우.} \\ M_{ij}, & \text{그렇지 않을 경우.} \end{cases}$ 
    end
  end
end

```

Step 3 : 이전의 코드를 이용하여 모든 M들을 계산한 후에, 모든 이용가능한 학습 패턴을 고려하면 규칙 결과들은 다음의 의사 코드에 따라 이들 요소로부터 결정 된다.

```

For i = 1, 2, . . . , n1 × n2
  집합 Mi {[Mij : j = 1, 2, . . . , n3]}로부터 최대
  매칭 요소 Mmax를 알아낸다.
  Mmax에 대응하는 term-node index(jmax)를 알아
  낸다.
  지수 jmax의 term 노드에 대하여 그것을 연결하
  는 것들은 제외하고 i번째 규칙 노드의 총 4개의
  연결을 모두제거한다.
end
  
```

Step 4 : 위의 알고리즘으로부터 출력의 언어 변수들의 항목 집합 중 오직 하나만이 어떤 일정한 퍼지 결과가 될 수 있다. 규칙 노드의 모든 매칭 요소들이 작으면 이것은 이 규칙이 출력에 영향이 적거나 거의 못 끼침을 의미한다. 이 경우 모든 대응 연결은 지워버리고 이 규칙 노드는 제거시키도록 한다.

3. 학습 3단계

소속 함수의 변수의 최적화에 대하여 GA를 사용한다. 문제에 따른 특정 지식은 GA가 학습 단계의 필요에 따라 맞추어서 사용한다. 제안된 접근법의 주요 기여도는 최적화 과정이 실행되는 동안에 퍼지 모델을 동적으로 수정할 수 있다는 것이다. 따라서 GA는 세대의 진행과 모델 형태의 변화를 가진 탐색 공간에서 변화를 한다. GA 탐색 공간은 단조적으로 점점 더 좁아지며 그 동안에 모델 변수들은 최적치에 점점 더 근접하게 된다.

제안한 GA는 모델 변수를 코드화하기 위하여 십진 정수 스트링을 사용한다. 십진 스트링은 퍼지 모델의 최적 변수를 결정하는데 있어서 적절한 표현방법이다. 대립 유전자의 수는 입력 및 출력 변수를 공간을 분배하는데 사용하는 퍼지 집합의 총 수로부터 결정될 수 있다. 그림 1에서 보인 것처럼 모델 형태처럼 소속함수($n_1+n_2+n_3=n_4$)를 가진다. 각각의 가우시안 형태의 소속 함수 중심 m 과 표준편차 σ 로 정의된다. 소속 함수를 최적화하기 위하여 ($n_4 \times 2$)변수를 최적화 하여야

한다.

이와 같이 GA는 길이 ($n_4 \times 2$)의 대립 유전자 스트링을 사용한다. 각 대립 유전자는 집합[1, 2, . . . , n_4]에서 어떤 값을 취할 수 있다.

대립 유전자의 값을 어떤 소속 함수의 새로운 중심과 표준편차로 변화하기 위하여 다음과 같은 절차를 수행한다.

Step 1 : 퍼지 제어기의 초기 중심값과 표준편차 값을 GA프로그램에 넣는다. 예를 들면, ($m_{i0} | i = 1, 2, \dots, n_4$)와 ($\sigma_{i0} | i = 1, 2, \dots, n_4$)이다.

Step 2 : 새로운 중심값과 표준편차가 대립 유전자 값으로부터 다음과 같이 계산 되어진다.

$$m_i = m_{i0} + (s_i - 5) * \delta_m \quad (6)$$

$$\sigma_i = \sigma_{i0} + (s_{(i+n_4)} - 5) * \delta_\sigma \quad (7)$$

여기서 m_i 와 σ_i 는 각각 새로운 중심과 표준편차 값이다. S_i 는 스트링에서 i 번째 대립 유전자의 값이다. δ_m 과 δ_σ 는 중심과 표준편차의 오프셋이다.

Step 3 : 어떤 중심이나 표준 편차의 값 s_i 가 5와 같다면 아무 변화도 발생하지 않는다. 만일 s_i 가 5보다 크다면 양의 변화가 일어나 중심이나 표준편차가 증가한다. 만일 s_i 가 5보다 작다면 음의 변화가 일어나서 중심 또는 표준편차가 감소한다.

제안한 GA는 적합도 함수로써 평균 제곱 오차 (Mean Square Error : MSE)를 사용한다. 퍼지 모델에 의해서 실제 출력과 목표 출력 사이의 차가 에러이다. 간단하게 각 염색체에 대하여 ($1/MSE$)는 염색체의 적합도 측정치로 사용될 것이다. MSE는 N개의 자료점으로부터 계산되어진다.

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (8)$$

여기서 y_i 는 실제값이고, \hat{y}_i 는 목표치이다.

각 세대 τ 에서 오프셋 값(δ_m 과 δ_σ)은 다음의 감쇠 함수에 의해 수정 되어진다.

$$\delta_m = \delta_m \times \theta_m \quad 0 < \theta_m < 1 \quad (9)$$

$$\delta_o = \delta_o \times \theta_o \quad 0 < \theta_o < 1 \quad (10)$$

여기서 θ_m 과 θ_o 는 각각 중심과 표준편차의 수정 값이 된다. 감쇠 함수는 지수 함수처럼 어떠한 감쇠 형태도 얻어낼 수 있다. 보통 GA프로그램의 종료 조건은 최대 허용 가능한 세대수 혹은 어떤 일정한 MSE의 값에 도달하는 경우로 정하고 있다. 본 논문의 GA 알고리즘의 경우에 종료 기준은 최적의 값에서 어느 정도 향상없이 어떤 큰 세대수에 도달하는 경우이다.

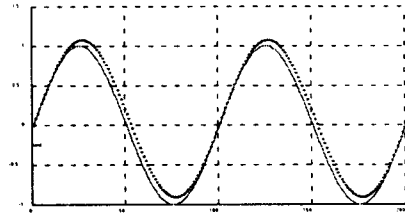
제안한 GA 프로그램은 모델 변수가 그들의 최적치에 도달할 때 탐색 공간의 해상도 내에서 동적 증가를 허용한다. 또한 정적 타입으로부터 동적 타입으로까지 GA의 너무 성급한 수렴의 기회를 최소화시킴으로써 모델 식별 문제의 성향도 변화시키며, 역전파 알고리즘(BP)이상의 장점을 가진다. GA는 BP가 받아들이지 않는 중간의 해결점을 받아들인다는 것이다. 또한 GA는 BP가 같은 조건하에서 수렴 문제가 제시되지만, 이러한 문제점은 없다.

V. 시뮬레이션

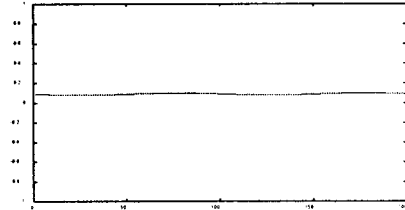
본 연구에서 제안한 알고리즘의 유용성을 확인하기 위해서 식 (11)로 표현되는 단일 링크 매니퓰레이터에 대하여 시뮬레이션 한다.

$$mi^2\ddot{\theta} + \delta\dot{\theta} + mgl\cos(\theta) = \tau \quad (11)$$

여기서 θ , $\dot{\theta}$, $\ddot{\theta}$ 는 각각 매니퓰레이터의 각변위, 각속도, 각가속도를 나타내고, τ 는 매니퓰레이터의 관절에 가해지는 토크이며, m , l , δ , g 는 각각 매니퓰레이터의 질량, 길이, 마찰계수, 주력가속도를 나타낸다. 이들의 값은 각각 $m = 0.2[\text{Kg}]$, $l = 0.15\text{m}$, $\delta = 0.4[\text{Kg} \cdot \text{m}^2/\text{s}]$, $g = 9.8[\text{m}/\text{s}^2]$ 의 사양을 갖는다. 그림 2는 단일 링크 매니퓰레이터의 초기값이 $\theta(0)=0$, $\dot{\theta}(0)=0$ 인 조건에서 $\hat{y}_i = \sin(t)$ 의 원하는 기준 궤적에 대한 퍼지-신경 회로망만 적용시켰을 경우의 추적 제어와 이에 대한 오차 곡선이다.



(a) tracking control($\theta(0)=0$, $\dot{\theta}(0)=0$)

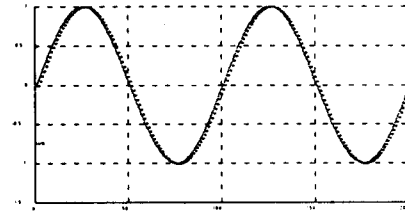


(b) tracking error

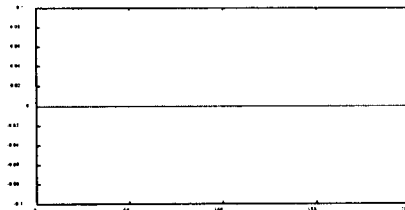
그림 2. 퍼지-신경 회로망의 추적 제어

Fig 2. The tracking control of Fuzzy-Neural Network

그림 3은 그림 2의 경우와 동일한 조건에서의 제안한 방식의 추종제어와 오차 곡선이다.



(a) tracking control($\theta(0)=0$, $\dot{\theta}(0)=0$)



(b) tracking error

그림 3. 제안된 방식의 추적 제어

Fig 3. The tracking control of proposed Network

그림 2와 그림 3에서 보듯이 기존의 퍼지-신경 회로망에 GA를 적용한 것이 추적 제어 성능이 더 좋음을 알 수 있다.

VI. 결론

본 논문에서는 GA를 이용한 비선형 시스템의 퍼지-신경 회로망 모델링을 제안하였다. 제안된 방식에서는 SOM을 이용하여 퍼지 소속 함수의 초기 상태를 구하며, 그 이후에 GA를 이용하여 퍼지 소속 함수의 중심과 표준 편차를 최적화시켰다. 비선형 시스템인 단일 링크 매니퓰레이터를 제어 대상으로 시뮬레이션 한 결과, 기존의 퍼지-신경 회로망을 쓰는 것보다 향상된 추적 제어 성능을 보여줌을 알 수 있다.

VII. 참고문헌

- [1] K.S. Narendra, K. Parthasarathy, "Identification and Control of Dynamic Systems using Neural Networks", IEEE Trans. on Neural Networks, Vol. 1, No. 1, pp. 4-27, March 1990.
- [2] M. Strefezza and Y. Dote, "Fuzzy and Neural Networks Controller", IECON'91, pp. 1437-1442, 1991.
- [3] D. Park, A. Kandel, and G. Langholz, "Genetic-Based New Fuzzy Reasoning Models with Application to Fuzzy Control", IEEE Trans. on Systems, Man, and Cybernetics, Vol. 24, No. 1, pp. 39-47, Jan 1994
- [4] Y.M.Chen, K.F.Gill, "Fuzzy-Neuro Control of Robotic Manipulators", The Third International Conference on Automation, Robotics and Computer Vision, pp. 1729-1733, 1994.
- [5] D. A. Linkens and J. Nie, "Back-propagation Neural-Network based on Fuzzy Controller with a Self-tuning teacher", INT. J. CONTROL, Vol. 60, No. 1, pp. 17-39, 1994.
- [6] J.S.R Jang and C.T. Sun, "Neuro-Fuzzy Modeling and Control", Proceedings of the IEEE, Vol. 83, No. 3, pp.378-406, March 1995.
- [7] A. Homaifar and E. McCormick, "Simultaneous Design of Membership Functions and Rule Sets for Fuzzy Controllers Using Genetic Algorithms", IEEE Trans. on Fuzzy

Systems, Vol. 3, No. 2, May 1995.

- [8] H. Ishibuchi, K. Nozaki, N. Yamamoto, and H. Tanaka, "Selecting Fuzzy If-Then Rules for Classification Problems Using Genetic Algorithms", IEEE Trans. on Fuzzy Systems, Vol. 3, No. 3, pp. 260-270, August 1995.