퍼지논리와 신경망 융합에 의한 로보트매니퓰레이터의 지능형제어 시스템 개발

On Developing The Intelligent Control System of a Robot Manipulator by Fussion of Fuzzy Logic and Neural Network

김 용 호*, 전 홍 태**
Yong Ho Kim and Hong Tae Jeon

※이논문은 1994년도 한국 학술 진흥 재단의 공모과제 연구비에 의하여 연구되었음

요 약

로보트 매니퓰레이터는 고도의 비선형 시변 시스템으로써 정밀한 제어가 매우 어려운 제어 대상으로 인식되어 왔으며 따라서 수많은 제어이론의 적용대상이 되어왔다. 로보트 매니퓰레이터의 제어에는 두가지 형태가 있는데 한가지는 궤적계획이고, 또 한가지는 궤적 추종이다. 본 논문에서는 궤적 추종을 목적으로 하고, 이를 위해 퍼지논리와 신경회로망을 결합한 지능형 제어를 제안한다. 제안된 제어시스템은 사고(思考) 및 추론(推論)과 같은 인간의 인식(認識) 처리에 해당하는 불확실한 것들의 구체화를 가능케하는 퍼지논리와 학습 및 병렬처리능력이 있는 신경회로망을 융합하여 구성된 퍼지-신경망 제어시스템이다. 그러나 이러한 장점을 갖는 퍼지-신경망 제어기도 정확한 제어 규칙의 발생은 어려운데 이는 신경회로망의 지역적 최소치에 빠지는 특성에 기인한다고 볼 수 있다. 그리고 일반적으로 시스템의 비선형 정도는 탐색에 의해서만 알수있는 성질의 것이므로 본 논문에서는 최적의 탐색알고리듬으로 널리 인정되고 있는 유전알고리듬을 사용하여 전역적인 규칙공간을 탐색한후 이를 바탕으로 퍼지-신경망 제어기를 완성한다. 제안된 제어시스템의 효율성은 2 자유도의 로보트 매니퓰레이터를 사용하여 컴퓨터 모의실험을 통해 입증된다.

ABSTRACT

Robot manipulator is a highly nonlinear-time varying system. Therefore, a lot of control theory has been applied to the system. Robot manipulator has two types of control ; one is path planning, another is path tracking. In this paper, we select the path tracking, and for this purpose, propose the intelligent controller which is combined with fuzzy logic and neural network. The fuzzy logic provides an inference morphorlogy that enables approximate human reasoning to apply to knowledge-based systems, and also

*정회원 : 중앙대학교 전자공학과 박사과정
**정회원 : 중앙대학교 전자공학과 부교수, 공학박사

provides a mathematical strength to capture the uncertainties associated with human cognitive processes like thinking and reasoning. Based on this fuzzy logic, the fuzzy logic controller(FLC) provides a means of converting a linguistic control strategy based on expert knowledge into automatic control strategy. But the construction of rule-base for a nonlinear time-varying system such as robot, becomes much more complicated because of model uncertainty and parameter variations. To cope with these problems, a auto-tuning method of the fuzzy rule-base is required. In this paper, the GA-based Fuzzy-Neural control system combining Fuzzy-Neural control theory with the genetic algorithm(GA), that is known to be very effective in the optimization problem, will be proposed. The effectiveness of the proposed control system will be demonstrated by computer simulations using a two degree of freedom robot manipulator.

I. 서 론

로보트 매니퓰레이터는 산업용, 교육용, 특수 목적용 등으로 개발되고 있고 널리 사용되고 있지만, 시스템이 고도의 비선형 시변특성을 갖고 있기 때문에 정밀한 제어가 매우 어려운 실정이다. 따라서 수많은 제어이론이 적용되고 여러 형태의 제어 시스템이 설계되어 왔지만 기존의 방법들은 정확한 수학적 모델링을 필요로하는 제약 때문에 정밀제어가 어렵거나 또는 불가능 했었다. 그리고 제어환경에 변화가 발생하였을 경우에는 제어시스템의 안정도가 보장되지 않는 경우마저 발생하였다.

최근에는 이러한 문제점들을 해결하기 위해 적응능력, 학습능력, 추론능력등을 갖는 지능형 제어기법이 개발, 적용되어 좋은 실행 특성을 보여주고 있다. 특히 퍼지논리와 신경회로망등은 대표적인 이론이라 할 수 있다.

그러나 퍼지논리 제어기 설계에 있어 가장 어려운 점은 규칙베이스를 구성하는 일이다. 전문가에 의한 규칙베이스의 구성은 상당히 주관적이고 경험에 의존하기 때문에 최적의 규칙베이스를 결정하기란 쉽지 않다. 특히 고도의 비선형 동특성(nonlinear dynamics)을 갖는 시스템의 제어 규칙을 단순히 전문가의 경험적 지식이나 실험에 의해 구성하는 일은 매우 어려우며, 많은 실행착오를 예상케 한다.

따라서 규칙베이스의 자동생성을 위한 방법에 대한 연구가 활발하게 이루어지고 있으며, 이러한 대표적인 연구결과로는 자기동조(self-tuning)방식[4], 학습능력을 갖는 신경망이론과의 융합방식[5][6][7]과 유전 알고리듬에 의한 방식[8][9]등이 있다.

자기 동조 방식은 실행 측정표(performance index)를 참고하여 현재 사용된 제어규칙의 기여도에 따라 제어규칙의 후건부를 강화 또는 약화시켜 제어규칙을 수정하는 방식이다. 그러나 실행 측정표를 정해주는 기준 역시 전문가의 경험과 지식에 의존하게 되므로 효율적인 방식이라 할 수 없다.

본 논문에서는 상기와 같은 문제점을 해결하기 위해서 최근에 최적화 알고리듬으로 널리 알려진 유전알고리듬과 퍼지-뉴럴 제어이론을 결합하여 퍼지-뉴럴 제어기의 퍼지소속함수와 제어규칙을 자동조정할 수 있는 방법을 제시하고, 이를 응용한 제어시스템을 제안한다. 유전 알고리듬은 해공간(solution space)에서 단일 해가 아닌 해집단(population of solution space)을 이용하기 때문에 전역적(global) 해의 발견을 가능케 하고, 최적화 함수의 정보(미분가능, 연속성)를 필요치 않는 장점을 갖는다. 그러나, 유전 알고리듬은 진화과정에서 최적점 근처에서의 미세조정(fine tuning)을 위해 많은 시간이 소비되고, 또한 off-line 학습 특성으로 인해 외부환경의 변화나 시스템 파라미터 변동에 대한 적응성의 문제가 대두된다.

따라서 제안된 퍼지-뉴럴 제어기는 제어대상 시스템의 역동력학 특성(inverse dynamics)의 퍼지 모델링(fuzzy modeling)에 기초하여 다음과 같은 2단계 학습을 통해 구성된다. 첫번째 단계는 오프라인 동조 과정(off-line course tuning) 단계로 유전 알고리듬을 사용하여 퍼지-뉴럴 제어기의 초기 파라미터(전건부 소속함수와 가중치로 표현되는 후건부 파라미터)를 추정하고, 두번째 단계는 온라인 미세동조 과정(on-line fine tuning) 단계로 이들 파라미터의 미세조정 및 시스템 파라미터 변동이나 외부환경의 변화에 대한 적응성을 부여하기 위해 오차 역전파 학습 알고리듬을 사용하여 제어환경에 적합한 퍼지-뉴럴 제어기의 최적의 파라미터를 추정한다.

본 논문의 구성은 다음과 같다. Ⅱ장에서는 퍼지-뉴럴제어기 구성이, Ⅲ,Ⅳ장에서는 유전 알고리듬과 이에 기초

한 1단계 과정이 설명되고, V장에서는 미세조정과정이, VI장에서는 퍼지-뉴럴 제어기가 설계 되고 VII장에서는 제안된 방식의 효율성을 입증하기 위하여 2자유도 평면 로보트 매니퓰레이터를 제어대상으로 모의실험을 수행하고 마지막으로 VIII장에서는 결론을 기술한다.

II. 퍼지-뉴럴 제어기 구성

본장에서는 퍼지-뉴럴 제어이론을 결합하여 퍼지소속함수와 제어규칙을 자동 조정할 수 있는 방법을 제시한다. 조정방법은 1 단계로 유전알고리듬에 의존하여 규칙의 전역적 영역을 찾기 위한 오프라인 동조 과정(off-line course tuning)을 거쳐 오차 역전파 알고리듬에 의한 온라인 미세동조 과정(on-line fine tuning) 및 적응학습의 2단계로 수행된다.

제안된 퍼지-뉴럴 제어기는 (식 1)과 같은 퍼지규칙(fuzzy rule)과 퍼지추론과정(the process of fuzzy reasoning)을 그림2와 같은 신경망(neural network) 구조로 구현하여 구성된다. 이러한 퍼지-뉴럴 제어기는 추론과정에 따라 전건부(premise part)와 후건부(consequence part)로 나누어진다. 퍼지-뉴럴 제어기구성을 위해 사용된 퍼지규칙과 퍼지추론과정 및 이를 구현한 신경망구조는 다음과 같다.

퍼지추론 방법은 혼합법을 선택하였는데 이 방법은 Takaki와 Sugeno[14]가 제안한 바 있으며 후건부 소속함수를 입·출력 선형관계식이나 상수값으로 표현하는 경우로써 비퍼지화기를 추론기에 포함시킨 형태이다. 이러한 혼합추론방식은 입력공간을 퍼지분할(fuzzy partition)하여 각각의 퍼지 부공간(fuzzy subspace)에 대한 선형 입출력 관계를 할당함으로써 기존의 선형 근사화 방식(piecewise linear approximation)에 비해 적은수의 규칙으로도, 비선형 특성을 갖는 입출력 관계를 보다 정확하게 표현할 수 있는 장점을 갖는다.

이때 사용되는 퍼지규칙은 (식 1)과 같은 퍼지함의(fuzzy implication)로 표현된다.

$$R_j: IF\ x_1\ is\ A_{1j}\ and\ \cdots\ and\ x_n\ is\ A_{nj},$$
$$THEN\ y_j = f_j(x_1, \cdots, x_n) \tag{1}$$

여기에서, R_j는 j번째 규칙, $x_1, \cdots, x_n$은 입력, A_{ij}는 퍼지변수, y_j는 j번째 규칙의 출력이며, 이때 후건부 함수는

$$f_j(x_1, \cdots, x_n) = a_{0j} + a_{1j}x_1 + \cdots + a_{nj}x_n \tag{2}$$

과 같이 입력변수들의 선형함수로 표현된다 (여기에서 $a_{ij}(i=0, 1, \cdots, n)$는 상수).

그리고 상기의 내용을 신경회로망으로 구현하는 과정은 다음과 같다. 퍼지-뉴럴 제어기의 전건부는 그림2에서 (A)~(C)-층(layer)으로 표현된다. (A)층은 입력변수를 전체집합(universe of discourse)상으로 설정하기 위한 층이며, (B)층에서는 전건부의 각 입력변수에 소속함수(membership function)를 할당하여 입력이 주어졌을 때, 그 소속정도를 출력한다. 그리고 소속함수의 형태는 다음과 같이 정의된다.

$$A_{ij}(x_i) = \exp\left(-\frac{1}{2} \cdot \left(\frac{x_i - c_{ij}}{w_{ij}}\right)^2\right) \tag{3}$$

여기서, c_{ij}와 w_{ij}는 각각 소속함수의 위치와 소속함수의 넓이, 즉 퍼지정도(fuzziness)를 결정하는 설계 매개변수이다. 그림3은 소속함수의 형태를 보여주고 있다.

퍼지규칙의 전건부 진리값(truth value)은 (C)층의 출력으로부터 결정된다. 이때 입력공간(input space)은 (B)층의 소속함수에 의해 퍼지 분할(fuzzy partition)되어 각각의 퍼지 부공간(fuzzy subspace)에서의 퍼지규칙의 진리값은 product-operation 규칙에 따라 다음과 같이 계산된다.

$$\mu_j = \prod_{i=1}^{n} A_{ij}(x_i) \tag{4}$$

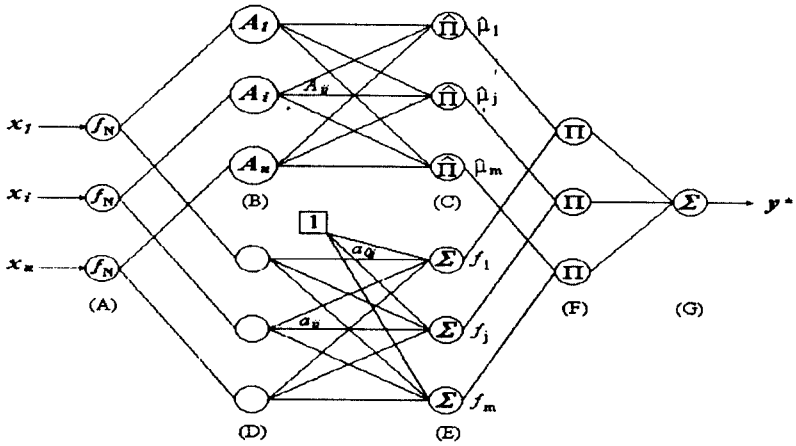


그림. 1. 퍼지-뉴럴 제어기의 구조
Fig. 1 The structure of Fuzzy-Neural controller

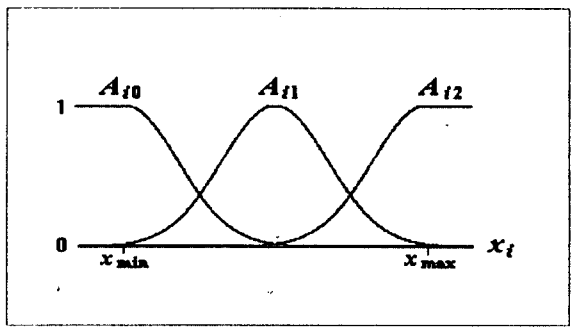


그림 2. 소속함수
Fig. 2 Membership functions

$$\hat{\mu}_j = \frac{\mu_j}{\sum_{k=1}^{m} \mu_k} \tag{5}$$

여기서, μ_j는 j번째 규칙 R_j의 진리값을 의미하며, $\hat{\mu}_j$는 μ_j의 정규화된 값으로 (C)층의 출력이다. 그리고 m은 규칙의 수로 (C)층의 유닛(unit)의 갯수에 해당한다.

퍼지-뉴럴 제어기의 후건부는 그림2의 (D)~(G)-층으로 표현되며, 실제 제어량 발생을 위한 최종 추론값을 계산한다. 식(1)의 퍼지규칙의 후건부 계수 aij는 (D)와 (E)층 사이의 가중치(weight)로 표현되며, 각 퍼지규칙의 추론값은 (E)층으로 부터

$$f_j(x_1,\cdots,x_n) = a_{0j} + a_{1j}x_1 + \cdots + a_{nj}x_n$$

과 같이 계산된다. 그리고 최종 추론값은 전건부의 진리값을 고려하여 (F), (G)층으로부터 다음과 같이 계산된다.

$$y^* = \frac{\sum_{j=1}^{m} \mu_j \cdot f_j(x_1,\cdots,x_n)}{\sum_{j=1}^{m} \mu_j} = \sum_{j=1}^{m} \hat{\mu}_j \cdot f_j(x_1,\cdots,x_n) \qquad (6)$$

여기서, 최종 추론값 y^*은 명확한 값(crisp value)으로 실제 제어 입력이 된다.

Ⅲ. 유전 알고리듬(Genetic Algorithms)

유전 알고리듬은 자연 생태계의 적응 매카니즘에 기초한 확률적(stochastic) 탐색 알고리듬으로 1970년대 John Holland에 의해 정립된 최적화 알고리듬이다.[10][11] 이러한 유전알고리듬은 최적화 문제, 분류 시스템, 신경회로망의 학습 문제[12][13], 퍼지 규칙베이스의 자동생성[17]등 여러 응용 분야에서 사용되고 있으며 다음과 같은 몇가지 특징을 갖고 있다.

첫째, 실제 파라미터가 아닌 부호화된 스트링, 즉 염색체(chromosome)를 이용하여 모의 진화를 수행한다.

둘째, 최적해의 탐색을 위해 해공간에서 단일해가 아닌 해집단을 이용함으로써, 전역적 해의 발견을 가능케 한다.

세째, 유전 알고리듬의 탐색과정은 확률적으로 일어난다.

상기와 같은 특징을 갖는 유전 알고리듬의 진화과정은 재생산(reproduction), 교배(crossover), 그리고 돌연변이(mutation)등의 연산자(operator)에 의해 처리되며, 일반적인 처리과정은 다음과 같이 요약할 수 있다.

[단계 1] 주어진 문제의 가능한 해를 부호화된 스트링으로 표현하여 초기집단을 구성

[단계 2] 집단의 각각의 스트링에 대한 적합도를 평가

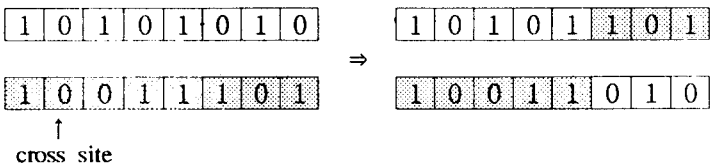
[단계 3] 적합도에 따라 현재 집단에서 스트링을 선택[재생산]

[단계 4] 선택된 어버이(parents) 스트링간의 교배 및 돌연변이의 진화과정을 통해 새로운 스트링 [자손(offsprings)]을 생성

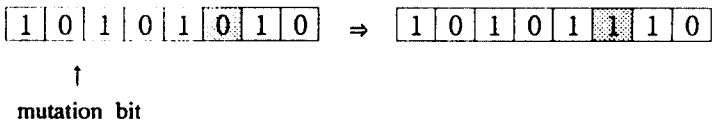
[단계 5] 새로 생성된 스트링에 대한 적합도를 평가하여 이전 세대와 교체

[단계 6] 최적해를 구할 때까지 3단계 이하의 과정을 반복

이러한 유전알고리듬의 수렴해석에 관한 연구는 스키마타(schemata) 이론에 기초하여 이루어지고 있으며, 유전알고리듬의 성능은 실제 파라미터의 부호화 기법, 재생산, 교배, 돌연변이 등의 유전 연산자(genetic operator)와 평가함수 설정 등에 크게 의존한다. 부호화 기법으로는 주어진 문제에 따라 여러 방법이 있지만, 일반적으로 이진 부호화 기법(binary coding technique)이 널리 사용되며, 이때 모든 스트링은 이진수로 표현된다. 재생산(reproduction)은 스트링의 선택(selection)과 관련되며, 스트링의 선택은 적합도(fitness)에 비례하여 확률적으로 일어난다. 즉, 높은 적합도를 갖는 스트링에 높은 선택 확률을 주어 다음세대를 구성한다. 적합도는 평가함수(evaluation function)에 의해 주어지며, 주어진 문제에 대한 스트링의 성능 척도(performance measure)가 된다. 교배(crossover)는 유전알고리듬의 진화과정에서 가장 중요한 부분으로 그 기능은 그림 3의 (a)와 같다. 이때 교배확률은 p_c이고, 교배점은 임의점으로 선택한다. 돌연변이(mutation)는 교배와 동시에 돌연변이 확률 p_c로 일어나며, 집단의 조기 수렴을 방지하는 역할을 하고, 그림 3의 (b)에서와 같이 임의 비트 전환으로 이루어진다.



(a) 교배(crossover)



(b) 돌연변이(mutation)

그림. 3. 교배와 돌연변이
Fig. 3 The crossover and the mutation operation

Ⅳ. 진화에 의한 퍼지분할과 규칙할당(Course Tuning)

제어환경에 적합한 규칙베이스를 구성하기 위해서는 입력공간의 퍼지분할을 위한 퍼지 소속함수와 각 퍼지부 공간에서의 규칙을 적절히 결정하여야 한다. 따라서 본 절에서는 그림1의 (B)층의 퍼지 소속함수와 (D), (E)층의 가중치로 표현되는 후건부 계수를 최적화 대상으로 하여 유전 알고리듬을 적용한다.

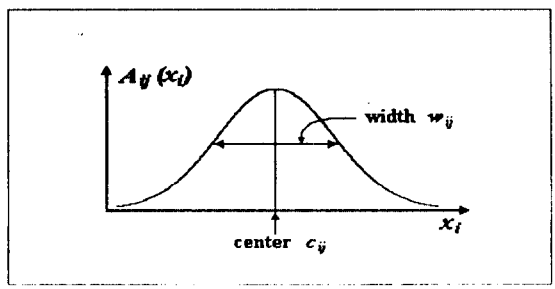


그림. 4. 소속함수의 파라미터
Fig. 4. Parameters of membership function.

각 입력변수에 할당된 퍼지 소속함수는 그림4와 같이 그 중심위치(center)와 폭(width)에 의해 특징 지울 수 있다. 따라서 이들 퍼지 소속함수의 파라미터(즉, c_{ij}와 w_{ij})와 후건부 계수(a_{ij})를 이진 부호화 하여 하나의 비트 스트링(bit string)을 구성한다.

A_{11}			A_{nq}			rule 1					rule m			
center	width	~	center	width		a_{01}	a_{11}	···	a_{n1}	~	a_{0m}	a_{1m}	···	a_{nm}
10001001	11001010		10101111	01011001		11001100~10011110					10101111~01011001			

그림. 5. 스트링 표현
Fig. 5. The string representation

그림5는 n개의 입력변수에 각 q개의 소속함수를 할당하였을 때의 비트 스트링을 보여준다. 이때, 규칙의 수 m은 q^n이 된다.

한편, 평가함수(evaluation function)는 상기의 스트링에 의해 표현되는 퍼지-뉴럴 제어기의 성능을 평가하는 기준이 되는 함수로, 집단내의 각 스트링에 대한 적합도를 측정한다. 따라서 최소의 오차와 오차변화율을 제어목적으로 하여 다음과 같은 평가함수를 선정한다.

$$F(s_i) = \frac{K_1}{Error} + \frac{K_2}{Change\ of\ Error} \tag{7}$$

여기서, s_i는 i번째 스트링이고, K_1, K_2는 상수이다.

상기의 스트링과 평가함수를 이용하여 진화에 의한 최적화 과정이 이루어진다. 본 논문에서는 정상상태 재생산 기법(steady-state reproduction technique)과, Roulette Wheel Parent Selection(RWPS) 알고리듬을 사용하였다. 정상상태 재생산 기법은 엘리트 전략(elitist strategy)의 하나로 세대교체(generational replacement)에 의한 재생산 기법과는 달리 집단에서 낮은 적합도를 갖는 임의의 n개 스트링을 새로 생성된 자손(offsprings)에 의해 교체하는 기법으로 이전세대의 높은 적합도를 갖는 스트링을 보존할 수 있다.

초기집단은 상기의 스트링 표현법에 맞추어 난수발생에 의해 구성되며, 집단내의 각 스트링은 식(7)의 평가함수에 의해 적합도가 결정된다. 이 집단으로부터 RWPS 알고리듬을 사용하여 적합도에 비례하여 스트링을 선택하고, 선택된 스트링은 교배와 돌연변이의 진화과정을 통해 새로운 스트링을 생성한다. 즉, 높은 적합도를 갖는 스트링에 높은 선택확률을 주어 이러한 스트링의 우수한 유전자를 새로운 스트링에 유전할 수 있는 확률을 높여준다. 그리고 새로운 스트링 즉 자손은 평가함수에 의해 적합도를 구하고 집단의 낮은 적합도를 갖는 스트링과 교체된다. 이와 같은 과정을 반복 실행하여 조정 매개변수의 최적점 또는 최적영역을 찾는다. 이때 교체되는 스트링의 수는 1개 혹은 2개로 설정한다. 교배(crossover)는 그림 1과 같이 두 스트링간의 교배점을 중심으로 한 교차로 일어나며, 교배확률 p_c는 1로써 선택된 스트링은 항상 교배가 이루어진다. 그리고 돌연변이는 앞의 Ⅱ장에서 설명한 것과 같이 임의로 정해진 확률 p_m으로 일어나며 비트 스트링의 임의 비트 전환으로 이루어진다. 이러한 돌연변이는 진화과정에서 조기에 지역적 최소점에 빠지는 것을 방지하는 역할을 한다. 이와 같은 진화를 통해 구성된 최종 집단에서 가장 높은 적합도를 갖는 스트링은 제어 목적에 가장 부합하는 스트링으로 퍼지-뉴럴 제어기의 초기 파라미터가 된다.

Ⅴ. 미세조정(fine tuning) 및 적응학습

본 절에서는 앞절의 유전알고리듬을 이용하여 추정한 퍼지-뉴럴 제어기의 매개변수들을 on-line 학습을 통하여 조정(tuning)하는 방법에 대해 설명한다. 학습 알고리듬은 역전파 학습알고리듬(backpropagation algorithm)[15]에 기초한 것이다. GFNC의 미세조정 과정은 다음과 같은 2단계로 수행된다. 먼저, 입력 데이터($x_1(t),\cdots,x_n(t)$)에 대한 실제출력($y^*(t)$)을 전방향 단계(forward step)를 통해 계산하고, 목표출력($d(t)$)과의 오차를 최소화하도록 역방향 단계(backward step)를 통해 조정 매개변수 즉, 전건부 소속함수(Gaussian membership function)의 위치 및 폭(width)과 후건부 선형식의 계수 등을 조정하는 단계로 수행된다. 출력층에서의 오차함수(E(t))는

$$E(t) = \frac{1}{2}(d(t) - y^*(t))^2 \tag{8}$$

과 같이 정의한다. 최급강하법(gradient descent method)을 이용하여 식 (8)의 오차함수 E를 최소화하는 매개변수의 조정량은 다음과 같이 결정된다.

전건부 소속함수의 위치(c_{ij})의 조정량은 반복규칙(chain rule)을 사용하여 다음과 같이 계산된다.

$$\Delta c_{ij} = -\eta \cdot \frac{\partial E}{\partial c_{ij}} = -\eta \cdot \frac{\partial E}{\partial y^*} \cdot \frac{\partial y^*}{\partial \mu_j} \cdot \frac{\partial \mu_j}{\partial w_{ij}}$$

$$= \eta \cdot (d - y^*) \cdot (f_j - y^*) \cdot \frac{\mu_j}{\sum_{k=1}^{m} \mu_k} \cdot \frac{(x_i - c_{ij})}{w_{ij}^2}$$

$$= \eta \cdot (d - y^*) \cdot (f_j - y^*) \cdot \hat{\mu}_j \cdot \frac{(x_i - c_{ij})}{w_{ij}^2} \tag{9}$$

그리고, 전건부 소속함수의 폭(w_{ij})의 조정량은 식(10)과 비슷하게 다음과 같이 계산할 수 있다.

$$\Delta w_{ij} = -\eta \cdot \frac{\partial E}{\partial c_{ij}} = -\eta \cdot \frac{\partial E}{\partial y^*} \cdot \frac{\partial y^*}{\partial \mu_j} \cdot \frac{\partial \mu_j}{\partial w_{ij}}$$

$$= \eta \cdot (d - y^*) \cdot (f_j - y^*) \cdot \hat{\mu}_j \cdot \frac{(x_i - c_{ij})^2}{w_{ij}^3} \tag{10}$$

또한, 그림의 (D), (E)층 사이의 가중치로 표현되는 후건부 입·출력 선형 관계식의 계수(coefficient)의 조정량은 다음과 같다.

$$\Delta a_{0j} = -\eta \cdot \frac{\partial E}{\partial a_{0j}} = -\eta \cdot \frac{\partial E}{\partial y^*} \cdot \frac{\partial y^*}{\partial f_j} \cdot \frac{\partial f_j}{\partial a_{0j}}$$

$$= \eta \cdot (d - y^*) \cdot \hat{\mu}_j \tag{11}$$

$$\Delta a_{ij} = -\eta \cdot \frac{\partial E}{\partial a_{ij}} = -\eta \cdot \frac{\partial E}{\partial y^*} \cdot \frac{\partial y^*}{\partial f_j} \cdot \frac{\partial f_j}{\partial a_{ij}}$$

$$= \eta \cdot (d - y^*) \cdot \hat{\mu}_j \cdot x_i \tag{12}$$

여기서 η는 학습률(learning rate)이고, i = 1,…,n, j = 1,…,m이다. 따라서, 퍼지-뉴럴 제어기의 매개변수는 최종적으로

$$c_{ij}(t+1) = c_{ij}(t) + \Delta c_{ij}(t) \tag{13}$$
$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t) \tag{14}$$
$$a_{0j}(t+1) = a_{0j}(t) + \Delta a_{0j}(t) \tag{15}$$
$$a_{ij}(t+1) = a_{ij}(t) + \Delta a_{ij}(t) \tag{16}$$

와 같이 갱신(update)된다. 여기서 t는 시간지수(time index)이다.

Ⅵ. 로보트 매니퓰레이터의 퍼지-뉴럴 제어시스템

앞의 Ⅲ장에서 제안한 GFNC를 이용한 로보트 매니퓰레이터의 퍼지-뉴럴 제어시스템 구조는 그림 6과 같다. 전체 제어시스템은 feedforward형의 제어구조로 병렬로 연결되어 있는 GFNC와 선형제어기, 그리고 제어 대상 시스템으로 구성되어 있다. 여기서 GFNC는 유전 알고리듬에 의한 off-line course tuning과 오차 역전파 알고리듬에 의한 온라인 미세조정 및 적응학습을 통해 제어대상 시스템의 역동력학 특성을 모델링하여 기준입력에 대한 제어 입력을 생성한다. 또한 선형제어기는 순간적인 제어환경의 변화 등에 대하여 시스템의 안정성을 보장하며, 또한 퍼지-뉴럴 제어기의 온라인 학습시 제어대상 시스템의 자코비안을 알 수 없는 경우에 간접적인 교사 신호를 제공하는 역할을 한다. 이것은 Kawato에 의해 제안된 피드백 학습방법으로 목표지향 학습방법(goal-directed

learning)이다.[15] 그리고 미세조정을 위한 식(9)의 오차함수는 선형제어기의 출력 u_{PD}에 의해서 다음과 같이 재정의 된다.

$$E=\frac{1}{2}(u_{PD}(t))^2 \tag{17}$$

그리고 제어 대상에 인가되는 전체 입력 u_{tot}는

$$\begin{aligned} u_{tot} &= u_{FN}(t)+u_{PD}(t) \\ &= u_{FN}(t)+K_p\cdot e(t)+K_d\cdot e(t) \end{aligned} \tag{18}$$

과 같이 선형제어기의 출력 u_{PD}와 GFNC 의 출력 u_{FN}의 합에 의해 결정된다. 여기서 $e(t)=r(t)-y(t)$이며, $r(t)$는 기준입력(reference input), $y(t)$는 시스템의 실제 출력이다.

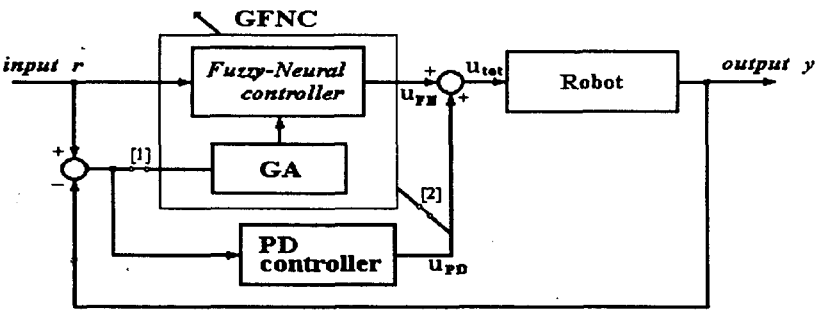
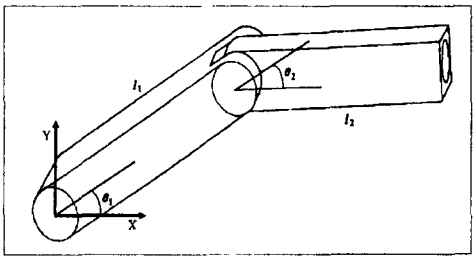


그림. 6. 퍼지-뉴럴 제어시스템
Fig. 6. Fuzzy-Neural control system

Ⅶ. 컴퓨터 모의실험 및 결과

본 논문에서는 제안된 시스템의 효율성을 보이기 위해 2 종류의 모의실험을 수행 하였다. 첫번째는 제안된 시스템의 구성 및 효율성 검증 실험이고, 두번째는 첫번째 실험에서 구성된 제어기의 적응성 검증실험이다.

제안된 제어시스템의 구성 및 검증을 위해 그림. 7과 같은 2 자유도 평면 로보트 매니퓰레이터의 위치제어를 선택하였다. 일반적인 로보트 매니퓰레이터의 동적 방정식은 식 (19)와 같이 표현된다.[16]



	Link 1	Link 2
Length	0.53 m	0.47 m
Mass	3.0 kg	2.5 kg

그림. 7. 자유도 로보트 매니퓰레이터
Fig. 7. degree of freedom robot manipulator

$$\tau = M(\theta)\ddot{\theta} + V(\theta, \dot{\theta}) + G(\theta) \tag{19}$$

여기에서, θ ($\in R^{n\times 1}$)는 관절 벡터, τ ($\in R^{n\times 1}$)는 토크 벡터, M(θ) ($\in R^{n\times n}$)는 관성행렬, $V(\theta, \dot{\theta})$ ($\in R^{n\times 1}$)는 원심력과 코리올리 힘벡터, G(θ) ($\in R^{n\times 1}$)는 중력 벡터이다. 식. 20과 같이 표현되는 로보트 매니퓰레이터는 다수의 비선형 항을 갖는 고도의 비선형 시스템으로 이를 위한 제어기 구성에 있어서 기존의 제어이론으로는 강건한 제어기 구성이 용이하지 못했다. 본 논문에서는 제어대상의 동적 방정식을 블랙박스로 취급하고, 상태변수는 관찰가능하다고 가정하여 모의실험을 수행하였다.

유전알고리듬에 의한 진화과정에서 퍼지-뉴럴 제어기의 각각의 파라미터는 8 비트 스트링으로 부호화 되어 하나의 비트스트링을 구성하며, 평가함수는 식 (20)의 평균자승오차(Mean Squared Error)와 오차변화분(Mean Squared Change of Error)에 의해 식 (7)과 같이 정의된다.

$$MSE = \frac{1}{T}\sum_{t=0}^{T}(\theta_d(t) - \theta_a(t))^2 \tag{20-a}$$

$$MSCE = \frac{1}{T}\sum_{t=0}^{T}(\dot{\theta}_d(t) - \dot{\theta}_a(t))^2 \tag{20-b}$$

여기서, θ_d와 θ_a는 각각 명령 궤적으로 준 관절 벡터와 시스템의 실제 출력이며, T는 명령 궤적의 한주기를 나타낸다. 또한 유전 알고리듬에서 집단수는 30개, 교배율은 100%, 돌연변이율은 1%로 주었고, 오차 역전파 학습 알고리듬의 학습률은 0. 0001로 하였다. 그리고 로보트 매니퓰레이터의 주행경로는 다음과 같은 직교좌표상에서의 타원형 궤적을 선정하였다.

$$X_d(t) = \begin{pmatrix} P_x(t) \\ P_y(t) \\ P_z(t) \end{pmatrix} = \begin{pmatrix} 0.3 + 0.25\cos\theta(t) \\ 0.3 + 0.15\sin\theta(t) \\ 0.0 \end{pmatrix} \tag{21}$$

모의실험 수행결과는 그림8~그림14에 주어져 있다. 그림 8은 초기 집단으로부터 진화과정에서의 적합도의 변화를 보여주고 있다. 이때 800번의 진화 후에는 거의 정상상태(steady state)로 수렴함을 알 수 있다. 그림 9, 10은 초기집단에서 가장 높은 적합도를 갖는 스트링에 의해 표현되는 퍼지-뉴럴 제어기에 의한 주행결과와 이때의 조인트 오차를 보이고 있으며, 그림 11, 12는 진화후의 로보트 매니퓰레이터의 주행결과와 조인트 오차를 보이고 있다. 그림 13은 진화후의 정상상태로 수렴한 파라미터를 퍼지-뉴럴 제어기의 초기 조건으로 하여 미세조정한 후의 주행결과를 보이고 있고, 이때의 조인트 오차는 그림 14와 같으며, 로보트 매니퓰레이터는 주어진 궤적을 거의 완벽하게 추종함을 알 수 있다. 표 3에서는 2000번 수행후의 본 논문에서 제안한 GFNC와 FNN과의 성능 비교를 보이고 있다.

표 1. GFNC와 FNN의 성능비교
Table 1. comparison of GFNC with FNN

FNN	Joint 1 (MSE)	5.9809E-05 (rad)
	Joint 2 (MSE)	3.3237E-05 (rad)
GFNC (without fine-tuning)	Joint 1 (MSE)	4.4345E-05 (rad)
	Joint 2 (MSE)	3.5727E-05 (rad)
GFNC (after fine-tuning)	Joint 1 (MSE)	1.8113E-05 (rad)
	Joint 2 (MSE)	0.7699E-05 (rad)

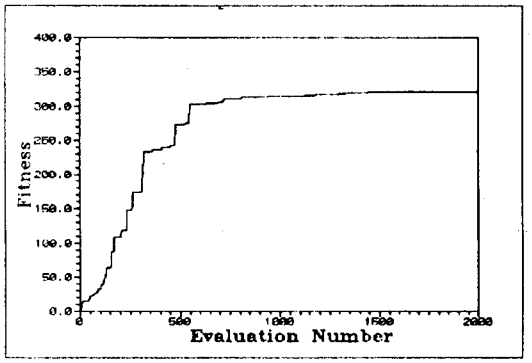


그림. 8. 진화과정에서의 적합도 변화
Fig. 8. Fitness variation in evolutionary process

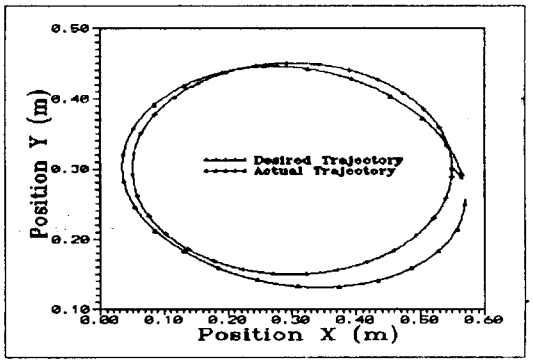


그림. 9. 초기집단에서 가장 높은 적합도를 갖는 스트링에 의한 주행결과
Fig. 9. The tracking result with initial population

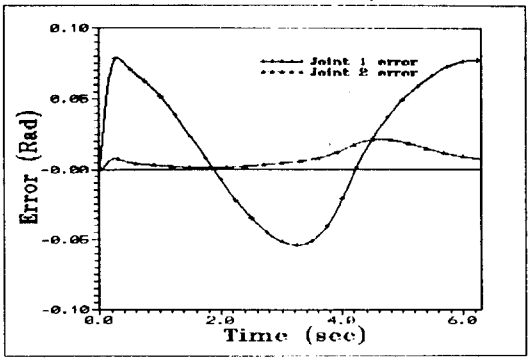


그림. 10. 초기집단에서 가장 높은 적합도를 갖는 스트링에 의한 조인트 오차
Fig. 10. The joint error with initial population

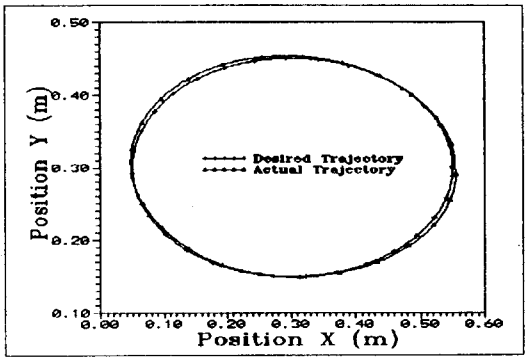


그림. 11. 진화후의 가장 높은 적합도를 갖는 스트링에 의한 주행결과
Fig. 11. The tracking result with best string after evolution

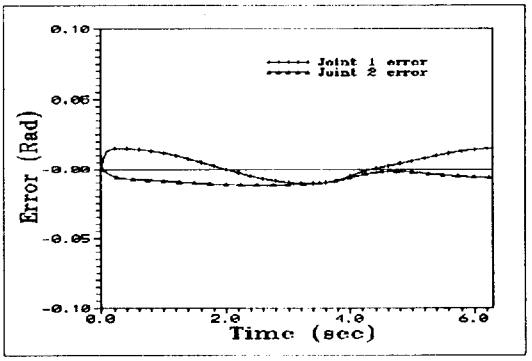


그림. 12. 진화후의 가장 높은 적합도를 갖는 스트링에 의한 조인트 오차
Fig. 12. The joint error with best string after evolution

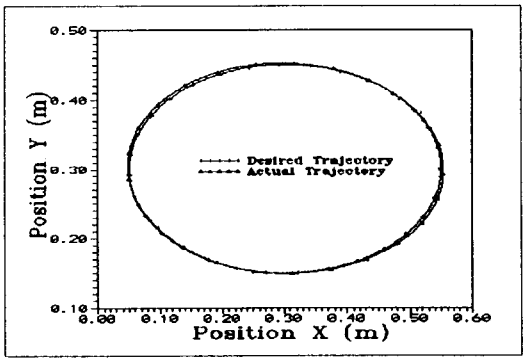


그림. 13. 미세조정 후의 주행결과
Fig. 13. The tracking result after fine-tuning

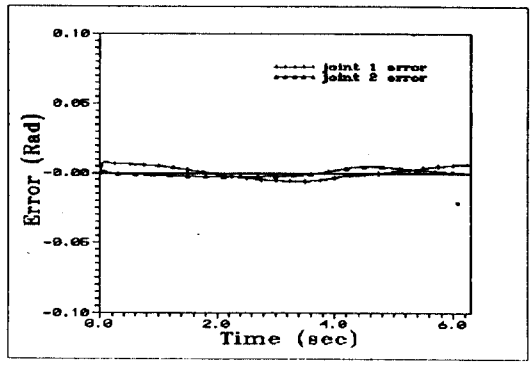


그림. 14. 미세조정 후의 조인트 오차
Fig. 14. The joint error after fine-tuning

Ⅷ. 결 론

퍼지-신경망 제어기는 추론의 장점을 갖는 퍼지논리와 병렬처리 및 학습의 장점을 갖는 신경회로망을 융합하여 구성된 제어시스템으로 매우 훌륭한 실행 특성을 갖는 시스템으로 평가받고 있다. 또한 제어대상에 대한 정확한 수학적 모델링이 요구되지 않으며 하드웨어로 구현할 경우 병렬처리 특성으로 인해 실시간 처리가 가능하다는 잇점을 갖는다. 그러나 제어대상이 고도의 비선형 동적 특성을 갖는 경우에는 전문가의 경험적 지식이나 신경회로망의 학습기능도 제어환경에 적합한 제어규칙의 전역적 영역을 찾는데 별로 도움이 되지 못한다.

따라서 본 논문에서는 제어환경에 적합한 최적의 규칙베이스의 구성을 위해 유전알고리듬과 퍼지-뉴럴 제어이론을 결합하여 퍼지 소속함수와 규칙베이스를 자동 조정할 수 있는 방법을 제시하고, 이를 응용한 로보트 매니퓰레이터의 제어시스템을 제안 하였다. 제안된 퍼지-뉴럴 제어기(GFNC)를 위한 조정 방법은 조정 매개변수의 최적영역의 탐색을 위해 유전알고리듬에 의한 전역적 영역 탐색 과정(course tuning)과 이들 매개변수의 미세조정과 제어환경의 변화에 유연하게 대응하기 위해 Gradient-type의 오차 역전파 알고리듬에 의한 미세조정(fine tuning)의 2단계 학습을 통해 수행되었다.

제안된 시스템은 각종 모델링 문제 및 타시스템에 직접적인 적용이 가능하며, 전문가 시스템, 패턴인식 시스템 등 타 분야로의 확장 적용이 가능할것으로 생각된다.

참 고 문 헌

1. M. Sugeno, M. Nishida, "Fuzzy control of model car," Fuzzy Sets and Syst., vol. 16. pp. 103-113, 1985.
2. J. A. Benard, "Use of a rule-based system for Process control," IEEE Control Sys. Mag., pp. 3-13, Oct. 1988.
3. Y. F. Li, C. C. Lau, "Development of fuzzy algorithms for servo systems," IEEE Control Sys. Mag., pp. 65-72, Apr. 1989.
4. S. Shao, "Fuzzy Self-Organizing Contoller and its Application for Dynamic Processes," Fuzzy Sets and Syst. , vol. 26, pp. 151-164, 1988.
5. H. Takagi, "Fusion Technology of Fuzzy Theory and Neural Network-Survey and Future Directions," Int. Conf. on Fuzzy Theory & Neural Network(IIZUKA '90), vol. 1, pp 13-26, 1990.
6. S. Horikawa, T. Horuhasi, S. Okuma, Y. Uchikawa, "Composition Method of Fuzzy Neural Networks," Proc. of IECON '90, vol. 2, pp. 1253-1258, 1990.

7. Y. Dote, "Fuzzy and Neural Network Controller," Proc. of IECON '90, vol. 2, pp. 1315-1343, 1990.
8. M. Lee, H. Takagi, "Integrating Design Stages of Fuzzy Systems Using Genetic Algorithms," Proc. 2nd IEEE Int. Conf. Fuzzy Systems, pp. 612-617, 1993.
9. P. Thrift, "Fuzzy logic synthesis with genetic algorithms," Proc. of the fourth Int. Conf. on Genetic Algorithms, pp. 509-513, 1991.
10. David E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Addison-Wesley Publishing Co., 1989.
11. L. Davis, Handbook of Genetic Algorithm, Van Nostrand Reinhold, 1991.
12. V. Maniezzo, "Genetic Evolution of the Topology and Weight Distribution of Neural Networks," IEEE Trans. on Neural Networks, vol. 5, no. 1, pp. 39-53, January 1994.
13. P. J. Angeline, G. M. Saunders, and J. B. Pollack, "An Evolutionary Algorithm that constructs Recurrent Neural Networks," IEEE Trans. on Neural Networks, vol. 5, no. 1, pp. 54-65, January 1994.
14. T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," IEEE Trans. Syst. Man Cybernet., vol. 15, pp. 116-132, 1985.
15. M. Kawato, Y. Uno, M. Isobe and R. Suzuki, "Hierarchical neural network model of for voluntary movement with application to robotics," IEEE Control Syst. Mag., pp. 8-16, Apr. 1988.
16. J. J. Craig, Introduction to Robotics : Mechanics and Control-2nd ed, Addison-Wesley, 1989.
17. Y. H. Kim, H. T. Jeon et. al "Self-Organization of Fuzzy Rule Base Using Genetic Algorithm" Fifth IFSA World Congress. 881-886,1993.