

유전자 알고리즘을 이용한 FPNN 모델의 최적 동정에 관한 연구

A Study on Optimal Identification of Fuzzy Polynomial Neural Networks Model Using Genetic Algorithms

이인태, 박호성, 오성권

원광대학교 전기전자 및 정보공학부, 전라북도 익산시 신용동 344-2 570-749

In-Tae Lee, Ho-Sung Park, Sung-Kwun Oh

School of Electrical Electronic & Information Engineering, Wonkwang Univ., Iksan.
KOREA

E-mail : ohsk@wonkwang.ac.kr

요 약

본 논문은 기존의 퍼지 다항식 뉴럴 네트워크 (Fuzzy Polynomial Neural Networks ; FPNN) 모델을 이용하여 비선형성 데이터에 대한 추론을 제안한다. 복잡한 비선형 시스템의 모델동정을 위하여 생성된 GMDH 방법에 기초한 FPNN의 각 노드는 퍼지 규칙을 기반으로 구축되었으며, 층이 진행되는 동안 모델 스스로 노드의 선택과 제거를 통해 최적의 네트워크 구조를 생성할 수 있는 유연성을 가지고 있다. FPNN 각각의 활성노드를 퍼지다항식 뉴론 (Fuzzy Polynomial Neuron ; FPN)이라고 표현한다. FPNN의 후반부 구조는 입력력 변수 사이의 간략과 회귀다항식 (1차, 2차, 변형된 2차식) 함수에 의해 구현된다. 규칙의 전반부 멤버십 함수는 삼각형과 가우시안형의 멤버십 함수가 사용된다. 또한 유전자 알고리즘을 사용하여 각 노드의 부분표현식을 구성하는 입력변수의 수, 입력변수와 차수의 선택 동조를 통하여 최적의 Genetic Algorithms(GAs)을 이용한 FPNN모델을 설계하는 것이 유용하고 효과적임을 보인다.

1. 서론

최근들어, 쉽게 이해할 수 있을 뿐만 아니라 일반화와 근사화 능력을 갖춘 시스템의 모델을 구축하기 위한 연구가 계속 진행되고 있다. 그러나 현재의 개발된 모델들은 차원이 증가하면(즉, 변수들의 수가 증가한다면), 시스템을 모델링하는데 어려움을 겪는다. 특히, 시스템이 복잡하고 대규모의 구조인 경우에는 함수식으로 모델의 표현이 한정되지 않기 때문에 회귀 분석을 적용할 수 없는 경우가 있다.

이러한 문제를 풀기 위해, A. G. Ivakhnenko는 Group Method of Data Handling(GMDH) 알고리즘을 소개하였다.[1] 그러나 GMDH는 비교적 간단한 시스템에 대해서 매우 복잡한 다항식을 생성하는 경우가 있으며, 2입력, 2차 회귀다항식의 형태로 복잡한 비선형 시스템을 표현하는데는

어느 정도 한계가 있다. 또한 시스템 입력변수가 3입력 이하인 경우엔 효율적인 모델을 구축하기가 어렵다. GMDH의 이러한 단점을 극복하고자, Oh 등에 의해 자기구성 다항식 뉴럴 네트워크 (Self-Organizing Polynomial Neural Networks ; SOPNN) [2]와 자기구성 퍼지 다항식 뉴럴 네트워크(Self-Organizing Fuzzy Polynomial Neural Networks ; SOFPNN) [3]가 제안되었다. SOFPNN (또는 FPNN)의 구조는 퍼지 추론 규칙을 기반으로 한 활성노드 즉, 퍼지 다항식 뉴론(Fuzzy Polynomial Neuron ; FPN)들로 구성되어 있으며, 일반적인 다층 퍼셉트론 구조의 경우와 달리 미리 정해져 있지 않고 층이 진행되는 동안 모델 스스로 노드의 선택과 제거를 통해 최적의 네트워크 구조를 생성할 수 있는 유연성을 가지고 있다. 그러나 기존 FPNN 구조는 성능을

향상시키기 위하여, 노드의 입력변수의 수와 규칙 후반부 다항식 차수 그리고 입력변수는 설계자의 경험 또는 반복적인 학습을 통해 선호된 네트워크 구조를 선택하였으나, 최적의 네트워크 구조를 구축하는 데는 어려움이 내재되어 있었다.

본 논문에서는 FPNN을 최적화시키기 위해서 최적화 탐색 방법인 유전자 알고리즘[4, 5]을 이용하여 FPNN의 입력변수의 수와 이에 해당되는 입력변수 그리고 규칙 후반부 다항식의 차수를 탐색하여 최적의 퍼지 다항식 뉴럴 네트워크를 구축한다. 따라서 모델 구축에 있어서 유연성과 정확성을 가지며 객관적이고 좀 더 정확한 예측 능력을 가진 GAs 기반 FPNN 모델 구조를 구축할 수가 있다.

본 논문의 평가를 위해서 소프트웨어 공정 데이터인 Medical Imaging System(MIS) 데이터 [6]를 이용하여 제안된 모델의 타당성, 정확성 및 예측 성능 등을 기존 지능 모델들과 비교를 통해 그 우수성을 보인다.

2. 퍼지 다항식 뉴론 (FPN)

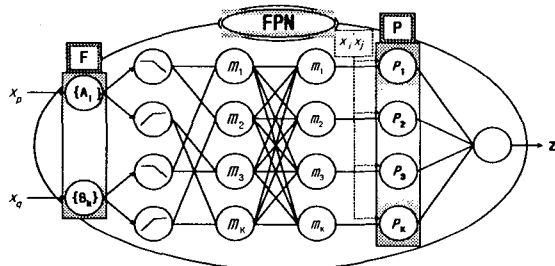


그림 1 FPN 모듈의 일반적인 구조

FPN은 FPNN의 가장 기본이 되는 처리 단위의 일반적인 형태로써, 퍼지 집합과 뉴럴 네트워크의 개념에 기초해서 자세히 설명된다. 그림 1에서 보여진 것처럼 FPN은 2개의 기본적인 기능 모듈로 구성되어 있다(여기서 그림 1은 예를 들어 2입력이고 멤버십 함수의 수는 2×2인 경우이다). 처음 하나는 “F”로 표기되어진 것으로, 이는 퍼지 집합의 집합체로써 입력 수의 변수들과 뉴론에 의해 실현된 처리 부분사이를 상호 연결해 준다. 그림에서 Xp와 Xq를 입력변수들로 표시한다. “P”로 표기된 두 번째 모듈은 함수 기반 비선형(다항식) 처리에 관한 것이다.

3. 유전자 알고리즘을 이용한 FPNN 설계

유전자 알고리즘을 이용한 자기구성 퍼지 다항

식 뉴럴 네트워크의 최적화 설계과정을 다룬다.

[단계 1] 출력변수 y에 관계하는 n개의 시스템 입력변수를 결정한다. 필요하다면 데이터를 정규화한다.

[단계 2] N개의 입출력 데이터를 학습용 데이터와 평가용 데이터로 분할한다.

[단계 3] GAs 기반 FPNN 구조를 구축하는데 있어서 기본적인 정보를 결정한다. 즉, GAs 기반 FPNN의 퍼지 추론 방법 및 동정을 설계하는 부분과 모델의 구조를 결정하는 초기 정보를 결정한다.

[단계 4] GAs 기반 FPNN 모델을 구축하는데 기본 유닛인 FPN 노드들을 유전자 알고리즘에서 각각의 염색체에 의한 구조 선택을 통해 최적 모델을 구축할 수가 있다.

[단계 4-1] 입력변수의 수 선택

[단계 4-2] 퍼지규칙 후반부 다항식 차수 선택

[단계 4-3] 입력변수 선택

[단계 4-1]부터 [단계 4-3]까지 수행해서 얻은 선택된 입력변수와 퍼지규칙 후반부 다항식 차수를 이용하여 퍼지 추론을 적용한다.

표 1 퍼지규칙 후반부에서 회귀다항식의 서로 다른 형태

입력수 차수	1	2	3
0(Type 1)	Constant		
1(Type 2)	Linear	Bilinear	Trilinear
2(Type 3)	Quadratic	Biquadratic	Triquadratic
2(Type 4)		Modified Biquadratic	Modified Triquadratic

퍼지 추론 규칙과 파라미터 동정을 위한 매개변수 추정은 다음과 같다.

i) 간략 추론

후반부가 단일의 상수항만을 가지는 것으로 이와 같은 추론법을 간략 추론법이라 한다.

$$R^i : \text{If } x_1 \text{ is } A_{i1} \text{ and } \dots \text{ and } x_n \text{ is } A_{in} \text{ then } y_j \text{ is } a_{j0} \quad (1)$$

i번째 데이터의 추론된 값은 다음과 같다.

$$\hat{y}_i = \frac{\sum_{j=1}^z w_{ji} y_j}{\sum_{j=1}^z w_{ji}} = \frac{\sum_{j=1}^z w_{ji} a_{j0}}{\sum_{j=1}^z w_{ji}} = \sum_{j=1}^z \hat{w}_{ji} a_{j0} \quad (2)$$

$$w_{ji} = A_{j1}(x_{1i}) \wedge \dots \wedge A_{jn}(x_{ni})$$

여기서, $i(i=1, \dots, m)$ 번째 데이터, m 은 데이터 총 수이다. 후반부 파라미터 a_{j0} 의 동정은 입력 데이터가 주어졌을 때 최소자승법에 의해 구해진다.

ii) 회귀다항식 추론

후반부가 1, 2, 변형된 2차 다항식 형태로 표현된 것으로 회귀다항식 추론이라 한다. 이 경우의 퍼지 모델은 다음과 같은 형태를 가지는 구현 규칙들로 구성된다.

$$R^j : \text{If } x_1 \text{ is } A_{j1} \text{ and } \dots \text{ and } x_n \text{ is } A_{jn} \text{ then } y_j = f_j(x_1, \dots, x_n) \quad (3)$$

[단계 5] 모델의 구축과 테스트 그리고 노드의 선택

Step 1) GAs 기반의 FPNN 모델 구축을 위한 유전자의 세대, Population 개수, 돌연변이를, 교배율, 그리고 염색체의 길이와 같은 유전자의 정보를 설정한다.

Step 2) 첫 번째 세대에서 설정된 population 개수만큼 염색체에 의한 노드(FPN)을 구축한다.

Step 3) 테스트 데이터를 이용해서 구해진 성능지수를 유전자 알고리즘의 적합도 함수로 계산하기 위해서 다음 식 (4)와 같이 변환하여 유전자 알고리즘의 적합도 함수로 사용한다.

$$F(\text{Fitness Function}) = 1/(1+\text{EPI}) \quad (4)$$

Step 4) 다음 세대의 생성을 위해서 구해진 개체구 정보와 적합도 값들을 이용하여 선택연산, 교배, 돌연변이를 실행한다.

Step 5) 구해진 적합도 값들(F_1, F_2, \dots, F_z)을 선호되는 적합도 값 순(내림차순)으로 정렬한다.

Step 6) 정렬된 적합도 값들 중에서 중복된 적합도 값들은 제거하고 하나의 적합도 값으로 처리한다.

Step 7) 중복된 적합도 값들을 하나로 처리한 후 설계자가 FPNN 구조를 설계하기전 결정된 노드의 제한된 개수 W개만큼 선호되는 적합도 값들을 선택한다.

Step 8) Step 7)에서 선택되어진 노드들을 다시 초기 population 번호 순서에 따른 오름차순으로 재정렬한다.

Step 9) 다음세대에서의 엘리트 전략을 위해서 선택된 노드들 중에서 최적의 적합도 값을 가지는 노드를 선택한다.

Step 10) Step 4)에서 얻어진 정보를 가지고 다음 세대의 population을 생성한 후 Step 2)에서 Step 9)까지 반복한다. 여기서 Step 3)에서 생성된 적합도 값들 중에서 가장 낮은 적합도 값을 가진 노드를 Step 9)에서 구한 이전 세대의 최적의 적합도 값을 가진 노드로 대체하는 엘리트 전략을 실행한다.

Step 11) 이전 세대에서 구해진 적합도 값들과 현재 세대에서 구해진 적합도 값들을 결합하여 다시 Step 5)부터 Step 8)까지 반복한다.

Step 12) 정해진 세대까지 Step 10)부터 Step

11)을 반복 실행한다.

[단계 6] [단계 5]의 현재 층에서 얻어진 최적 노드의 최대 적합도 값 F_1 이 다음 부등식을 만족하는 경우에 알고리즘을 종료한다.

$$F_1 \leq F_* \quad (5)$$

단, F_* 는 이전 층의 최적 노드인 최대 적합도 값이다. 본 논문에서의 모델의 성능지수식으로 다음 식을 이용하였다.

$$E = \frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2 \quad (6)$$

[단계 7] 현재 층에서 보존된 노드의 출력들($z_{1i}, z_{2i}, \dots, z_{wi}$)에서 다음 층의 새로운 입력들($x_{1j}, x_{2j}, \dots, x_{wj}$)로, $x_{1j}=z_{1i}, x_{2j}=z_{2i}, \dots, x_{wj}=z_{wi}$ 으로 구성하고, [단계 4]로 간다(여기서, $j=i+1$). 이 후, [단계 4]부터 [단계 7]까지를 반복한다. 알고리즘이 종료할 경우, 마지막 층의 최적 적합도 값 F_* 을 얻은 노드의 다항식에 입력이 되는 전 층의 출력을 대입하고 동일한 조작을 제 1층까지 반복하여 최종 추정 모델 \hat{y} 를 얻는다.

4. 실험 데이터를 통한 결과 고찰

제안된 모델의 성능을 테스트하기 위해서 MIS 데이터 [6]를 이용하였다. 이 공정은 기존의 이용할 수 있는 데이터들 보다 풍부한 데이터들로 구성되어 있기 때문에 모델을 평가하는데 많은 도움이 된다. 표 2는 유전자염색체 정보와 고급 퍼지 다항식 뉴럴 네트워크 구축을 위한 정보를 나타낸 것이다.

표 2 GA과 FPNN정보

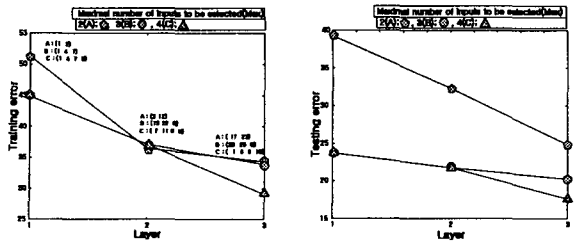
	Parameters	1층~3층
GA	Maximum gen	100
	Total population size	60
	Selected population size	30
	Crossover rate	0.65
	Mutation rate	0.1
	String length	3+3+30
FPNN	Maximal no. of inputs to be selected(Max)	$1 \leq l \leq \text{Max}(2 \sim 5)$
	polynomial Type(Type T) of the consequent part of rules	$1 \leq T \leq 4$
	Membership Function(MFs) type	1.Gaussian 2.Triangular
	No. of MFs per each input	2

l, T ; interger

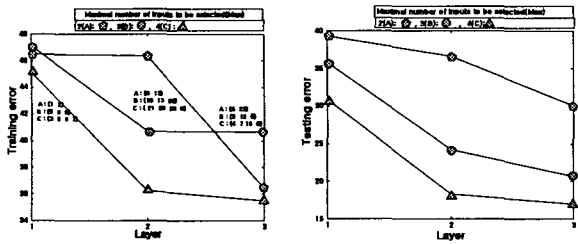
* ; 규칙 후반부 다항식의 입력이 전체 시스템 입력이 사용된 경우

그림 2는 선택된 입력의 변화에 따른 GAs 기반의 FPNN 각 층의 성능지수를 나타낸 것이다

그림 3은 선택된 입력변수의 수가 4입력이고 멤버십 함수로 가우시안 함수로 실제 데이터와 모델화한 데이터의 오차를 나타내고 있다.

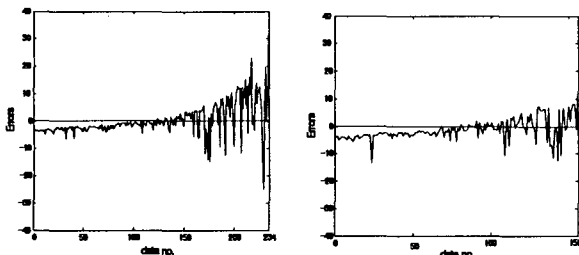


(a) Triangular MFs



(b) Gaussian-like MFs

그림 2 층의 증가에 따른 FPNN의 성능지수



(a) Training error (b) Testing error

그림 3 GAs 기반 FPNN의 오차곡선

표 3 기존의 모델과 동정오차비교

	Max	M_type	T	PI	EPI
FPNN [3]	2	T	2	34.169	25.997
		G	1	47.404	32.535
	3	T	1	34.637	21.753
		G	1	37.665	21.364
Our model	2	T	4	34.257	24.837
		G	4	36.432	30.071
	3	T	4	33.833	20.254
		G	1	40.677	20.694
	4	T	1	28.598	17.596
		G	1	35.436	16.947

T : Triangular , G : Gaussian-like

표 3은 MIS 데이터[6]를 사용하여 성능지수 식 (6)에 의해 계산된 값으로 기존의 모델과의 성능비교를 한 것이다. Max=4 이고 멤버십함수가 가우시안형이며 Type 1일 때 PI와 EPI성능이 우수함을 보인다. 또한 기존의 FPNN보다 Our model이 Max값과 M_type의 형태 그리고 차수

(T) 선택에 있어서 좀 더 유연성과 정확성을 가지고 예측능력이 우수함을 알 수 있다.

5. 결론

본 논문에서는 GAs 기반의 FPNN의 효율적인 설계방법을 제안하였다. 기존의 FPNN 구조는 성능을 향상시키는데 크게 영향을 미치는 네트워크 층, 노드내의 입력변수의 수와 퍼지규칙 후반부 다항식의 차수 그리고 고정된 입력변수 선택을 설계자의 경험에 의해서 결정함으로써 객관적이고 주어진 데이터의 특성에 맞는 최적의 모델을 구축하는데 어려움이 있었다. 이러한 FPNN 모델 구조의 한계를 극복하고자 본 연구에서는 최적화 방법인 유전자 알고리즘을 이용하여 노드의 입력변수의 수와 그에 해당하는 입력변수 그리고 후반부 다항식 차수 등의 파라미터를 선택하여 모델 구축에 있어서 유연성과 정확성을 가지며 더욱 정확한 예측 능력을 가진 최적화된 GAs 기반의 FPNN 모델 구조를 구축할 수가 있었다.

감사의글

본 연구는 산업자원부의 지원에 의하여 기초전력연구원(R-2003-B-274) 주관으로 수행된 과제임.

6. 참고문헌

[1] A. G. Ivakhnenko, "Polynomial theory of complex systems", *IEEE Trans. on Systems, Man and Cybernetics*, Vol. SMC-1, pp. 364-378, 1971.

[2] S.-K. Oh and W. Pedrycz, "The design of self-organizing Polynomial Neural Networks", *Information Science*, Vol. 141, pp. 237-258, 2002.

[3] S.-K. Oh and W. Pedrycz, "Fuzzy Polynomial Neuron-Based Self-Organizing Neural Networks", *Int. J. of General Systems*, Vol. 32, No. 3, pp. 237-250, May, 2003.

[4] Holland, J. H., *Adaptation In Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor. 1975.

[5] K. De Jong. Are genetic algorithms function optimizers? In Proc. of PPSN II (Parallel Problem Solving from Nature), pages 3-13, Amsterdam, North Holland, 1992.

[6] Michael R. Lyu, "Handbook of Software Reliability Engineering", *Computing McGraw-Hill*, pp. 510-514, 1996