

# 유전자 알고리즘에 의한 자기구성 다항식 뉴럴 네트워크의 구조 및 동정 알고리즘

## The Architecture and Identification Algorithm of Self-Organizing Polynomial Neural Networks by GAs

박호성, 오성권

원광대학교 전기전자및정보공학부

Ho-Sung Park, Sung-Kwun Oh

School of Electrical and Electronic Engineering, Wonkwang University

E-mail : ohsk@wonkwang.ac.kr

### 요 약

본 논문에서는 유전자 알고리즘에 기반을 둔 자기구성 다항식 뉴럴네트워크 (Self-Organizing Polynomial Neural Networks : SOPNN)의 새로운 구조를 제안하고, 포괄적인 설계 방법론을 토의한다. 기존의 자기구성 다항식 뉴럴 네트워크는 확장된 GMDH 방법을 기반으로 하며, 네트워크의 성장과정을 통하여 각 층의 다항식 뉴런에서 고정된 노드 입력들의 수뿐만 아니라 다항식 차수(1차, 2차, 그리고 수정된 2차식)를 이용하였다. 더구나, 그 방법은 학습을 통해 생성된 SOPNN이 최적 네트워크 구조를 가진다는 것을 보증하지 못한다. 그러나, 제안된 GA 기반 SOPNN은 그 구조를 구조적으로 더 최적화된 네트워크가 되도록 하고, 기존의 SOPNN보다 훨씬 더 유연하고, 선호된 뉴럴 네트워크가 되도록 한다. 구조적으로 더 최적화된 SOPNN을 생성하기 위해, SOPNN의 각 단계에서의 GA기반 설계 절차는 SOPNN내에서 이용할 수 있는 다음의 최적 파라미터들- 즉 입력변수의 수, 입력변수, 및 다항식 차수-을 가진 선호된 노드들의 선택으로 이끈다. 하중계수를 가진 합성성능지수가 그 모델의 근사화 및 일반화(예측) 능력 사이의 상호 균형을 얻기 위해 제안된다. 상세 설계 절차가 상세히 토의된다.

### 1. 서론

시스템이 복잡하고 대규모의 구조인 경우에는 함수식으로 모델의 표현이 한정되지 않기 때문에 회귀 분석을 적용할 수 없는 경우가 있다. 이러한 문제를 풀기 위해, A. G. Ivakhnenko[1]는 GMDH(Gro up Method of Data Handling) 알고리즘을 소개하였다. GMDH의 주된 특성은 자기 구성과 적자생존의 자연 법칙에 근간을 둔 진화적 알고리즘이다. 그러나 이는 2입력, 2차 회귀다항식의 형태로 복잡한 비선형 시스템을 표현하는데는 어느 정도 한계가 있다. 이러한 단점을 극복하고자 Oh et al. 등에 의해 자기구성 다항식 뉴럴 네트워크(Self-Organizing Polynomial Neural Networks : SOPNN)[2, 3]가 개발되었다. SOPNN의

구조는 GMDH의 2입력 2차 다항식에서 벗어나 다변수 다차식의 확장된 부분표현식을 이용하여 연속적인 층과 노드(Polynomial Neuron)를 생성함으로써 최적의 구조를 얻는 알고리즘으로 기존의 신경망 알고리즘보다 더 적은 노드수와 유연한 적응특성을 가지고 있다. 이러한 SOPNN 성능은 기존의 퍼지 모델링 방법보다 우수한 성능을 보여준다. 그러나 SOPNN 구조의 성능을 향상시키는데 크게 영향을 미치는 노드의 입력변수의 수와 차수 그리고 입력변수는 설계자의 경험에 의해서 결정함으로써 최적의 네트워크를 구축하기 위해서는 다양한 네트워크 구조와 그와 관련된 다양한 방법론적 설계가 필요하였다.

본 논문에서 제안된 자기 구성 다항식 뉴럴 네

트위크는 최적화 탐색 방법인 유전자 알고리즘 [4, 5]을 이용하여 자기구성 다항식 뉴럴 네트워크의 입력변수의 수와 이에 해당되는 입력변수 그리고 다항식의 차수를 탐색하여 최적의 자기구성 다항식 뉴럴 네트워크를 구축한다. 또한 하중 계수를 가진 목적함수[6]를 이용하여 모델의 근사화와 일반화 능력사이의 합리적 균형을 통하여 노드의 성능을 평가함으로써 상호 연계된 최적 모델을 구축할 수가 있다.

본 논문의 평가를 위해서 비선형 시스템 데이터[7-9]를 이용하여 제안된 모델의 타당성, 정확성 및 예측 성능 등을 기존 지능 모델들과 비교를 통해 그 우수성을 보인다.

## 2. 자기구성 다항식 뉴럴 네트워크의 구조와 동정 알고리즘

### 2.1 자기구성 다항식 뉴럴 네트워크

GMDH 알고리즘을 기반으로 SOPNN 설계 알고리즘은 다음과 같이 단계별로 설명된다.

[단계 1] 출력변수  $y$ 에 관계하는  $n$ 개의 시스템 입력변수를 결정한다.

[단계 2] 전체 입출력 데이터를 학습용 데이터와 테스트용 데이터로 분할한다.

[단계 3] SOPNN의 다양한 구조 선택을 통하여 주어진 시스템 환경의 변화에 적응할 수 있는 유연성을 가진 모델을 구축한다.

[단계 4] 선택한 모델에 따라 각 노드의 입력변수 수와 다항식 차수를 결정한다.

입력수 다항식차수	1	2	3
1(Type 1)	Linear	Bilinear	Trilinear
2(Type 2)	Quadratic	Biquadratic	Triquadratic
3(Type 3)		Modified	Modified
		Biquadratic	Triquadratic

표 1 회귀다항식의 차수 형태

[단계 5] 선형 회귀분석에는 학습 데이터를 사용하고, 성능지수 즉 동정오차는 APE(Average Percentage-based Error)를 사용한 식 (1)을 이용한다.

$$E = \frac{1}{N_t} \sum_{m=1}^{N_t} \frac{|y_m - \hat{y}_m|}{\hat{y}_m} \times 100\% \quad (1)$$

여기서,  $N_t$ 는 학습데이터 총 수이다.

추정된 후반부 다항식에 테스트 데이터를 사용하여 식 (1)의 테스트 데이터에 대한 동정오차를 계산한다. 얻어진 동정 오차값들에 대해서 2층 이상부터는 다음과 같이 값이 작은 순으로

배열한다.

[단계 6] 단계 5의 현재 층에서 얻어진 최소 동정오차  $E_i$ 이 다음의 부등식을 만족하는 경우에 알고리즘을 종료한다.

$$E_1 \geq E_* \quad (2)$$

단,  $E_*$ 는 이전 층의 최소 동정오차이다.

[단계 7] 현재 층에서 보존된 노드의 출력들에서 다음 층의 새로운 입력들로 구성되고, [단계 4]로 간다. 이 후, [단계 4]부터 [단계 7]까지를 반복한다. 알고리즘이 종료할 경우, 최소 동정오차  $E$ 를 얻은 노드의 다항식에 전 층의 출력을 대입하고 동일한 조작을 제 1층까지 반복하여 최종 추정 모델  $\hat{y}$ 를 얻는다.

### 2.2 유전자 알고리즘

유전자 알고리즘(GAs)은 유전학과 자연 진화를 흉내낸 적응 탐색법으로 1975년 Holland[4]에 의해 개발되었다. 본 논문에서는 유전자 알고리즘을 시뮬레이션하기 위해 설정된 각각의 개체에 존재하는 염색체의 수와 군집의 크기에 대한 파라미터의 정의으로써, 한 개체에 여러 변수의 정보를 담은 직렬 방식을 사용한다. 재생산연산으로 roulette-wheel 방식, 돌연변이연산은 선택된 비트를 반전시키는 invert 방식, 교차연산은 one-crossover 방식을 사용하였다. 그러나 roulette-wheel 방식은 확률적 속성 때문에 선택 과정에서 최적자를 반드시 선택하지 못하는 단점을 가진다. 이를 보완해줄 수 있는 한 방법이 엘리트 전략(Elitist strategy)이다[5]. 본 논문에서는 기존의 자기구성 다항식 뉴럴 네트워크가 가지고 있던 단점인 각 노드의 입력변수의 수와 이에 해당하는 입력변수 그리고 다항식 차수의 선택을 최적화 알고리즘인 유전자 알고리즘을 통해 탐색하여 경험자의 경험에 의존한 구조의 최적화를 객관적이고 효율적으로 구축하게 된다.

### 2.3 하중계수를 가진 목적함수

본 논문에서는 근사화 및 일반화의 상호 균형과 의존 능력을 가진 합성 목적 함수(성능지수)를 이용하여 상호 연계된 최적 모델 구조를 생성한다[6]. 이를 위해 학습 및 테스트 데이터의 성능 결과에 대한 하중계수를 가진 목적함수인 합성 성능지수으로써 식 (3)과 같이 정의한다.

$$f(PI, EPI) = \theta \times PI + (1 - \theta) \times EPI \quad (3)$$

하중계수를 가진 목적함수인 식 (3)을 유전자 알고리즘과의 연계를 위해서 유전자 알고리즘의 적합도 함수로 다음과 같이 식 (4)을 사용한다.

$$F(\text{Fitness Function}) = \frac{1}{1 + f(PI, EPI)} \quad (4)$$

$\theta$ 는  $PI$ 와  $EPI$ 에 대한 하중계수를 나타내고,  $PI$ 는 학습 데이터에 대한 성능지수,  $EPI$ 는 테스트 데이터에 대한 성능지수를 나타낸다.

### 3. 유전자 알고리즘에 의한 자기구성 다항식 뉴럴 네트워크 설계

유전자 알고리즘을 이용한 자기구성 다항식 뉴럴 네트워크의 최적화 설계방법은 다음과 같다.

[단계 1] 출력변수  $y$ 에 관계하는  $n$ 개의 시스템 입력변수를 결정한다.

[단계 2] 전체 입출력 데이터를 학습용 데이터와 테스트용 데이터로 분할한다.

[단계 3] SOPNN 구조를 구축하는 있어서 기본적인 정보를 결정한다.

[단계 4] 유전자 알고리즘을 이용한 SOPNN 구조 결정 및 입력변수 수와 회귀 다항식 차수 그리고 입력변수를 선택한다. 기존의 SOPNN 구조[3, 4]에서는 하나의 층에서 모든 노드가 모두 똑같은 입력변수의 수와 다항식 차수를 가지고 모델을 구축하지 못하는 단점이 있었다. 이에 비해 본 논문에서 제안한 유전자 알고리즘을 이용한 SOPNN 구조는 위에서 살펴본 그림 3과 같은 염색체 처리를 SOPNN 전체 층의 모든 노드에 각각 실행함으로써 기존의 SOPNN 구조의 일률적인 입력변수의 수와 다항식 차수에서 벗어나 모델을 최적화하기 위한 입력변수의 수와 그에 해당하는 입력변수 그리고 다항식 차수가 유전자 알고리즘에 의해 모델이 최적화되기 위해 선택되어진다.

[단계 5] SOPNN 모델의 구축과 테스트 그리고 노드의 선택을 한다. 기존의 SOPNN에서는 모델의 구축은 앞에서 설명하였듯이 일반적인 설계방법이었으나 본 논문에서는 유전자 알고리즘을 이용하여 모델의 최적화에 맞게 각 노드들을 유연하게 구축할 수 있다.

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

$$F_1 \leq F_s \quad (5)$$

단,  $F_s$ 는 이전 층의 최적 노드인 최대 적합도 값이다.

[단계 7] 현재 층에서 보존된 노드의 출력들 ( $Z_{1j}, Z_{2j}, \dots, Z_{Nj}$ )에서 다음 층의 새로운 입력들 ( $X_{1j}, X_{2j}, \dots, X_{Nj}$ )로,  $X_{1j}=Z_{1j}, X_{2j}=Z_{2j}, \dots, X_{Nj}=Z_{Nj}$ 으로 구성하고, [단계 4]로 간다(여기서,  $j=j+1$ ). 이 후, [단계 4]부터 [단계 7]까지를 반

복한다. 알고리즘이 종료할 경우, 마지막 층의 최적 적합도 값  $F_s$ 을 얻은 노드의 다항식에 입력이 되는 전 층의 출력을 대입하고 동일한 조작을 제 1층까지 반복하여 최종 추정 모델  $\hat{y}$ 를 얻는다.

### 4. 시뮬레이션 및 결과 고찰

제안된 자기구성 다항식 뉴럴 네트워크의 성능을 판단하기 위해서 3입력 1출력 비선형 함수 데이터[7-9]를 사용하였다. 사용된 비선형 함수는 다음 식 (6)과 같이 구성되며, 성능지수 식은 식 (4)을 이용한다.

$$y = (1 + x_1^{0.5} + x_2^{-1} + x_3^{-1.5})^2 \quad (6)$$

전체 40개 데이터 중 처음 20개 데이터는 학습 데이터로 나머지 20개 데이터는 테스트 데이터로 사용한다. 그림 1은 층의 증가에 따른 SOPNN의 성능지수를 나타낸 것이다.

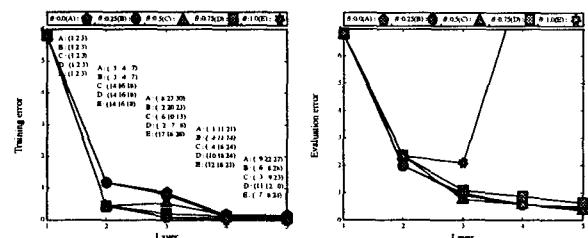


그림 1 층의 증가에 따른 SOPNN의 성능지수

그림 1에서 알 수 있듯이 층이 증가할수록 모델의 근사화와 일반화 능력이 뚜렷하게 향상되는 것을 알 수 있다. 비선형 데이터를 이용하여 제안한 방법으로 모델을 구축하였을 경우, 하중계수 값이  $\theta=0.5$ 일 때  $PI=0.124$ ,  $EPI=0.337$ 인 값을 얻을 수 있다. 그림 2는 하중계수 값이  $\theta=0.5$ 인 경우에 3층에서 네트워크를 나타낸 것이다.

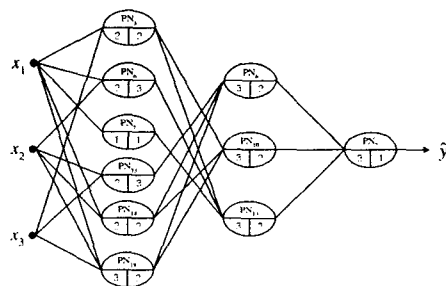


그림 2 최적의 SOPNN 구조( $\theta=0.5$ )

그림 3은 하중계수 값이  $\theta=0.5$ 인 경우에서 5층일 때의 주어진 데이터의 출력값과 모델의 출력값을 비교한 그림이다.

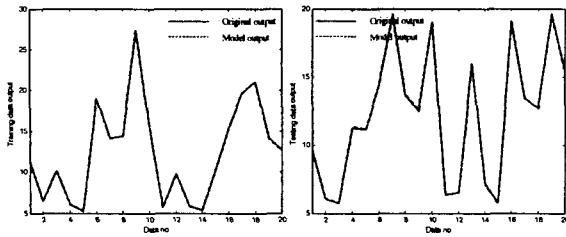


그림 3 비선형 함수 데이터의 출력값과 모델의 출력값

표 2 기존 모델방법과의 동정 오차비교

Model		Performance Index	
		PI	EPI
GMDH[10]		4.7	5.7
Linear model[9]		12.7	11.1
Shinichi's model[7]	Type 1	0.84	1.22
	Type 2	0.73	1.28
Sugeno's model[8]	Model I	1.5	2.1
	Model II	1.1	3.6
Our model	$\theta=0.5$	0.124	0.337

표 2는 3입력 1출력 비선형 함수 데이터를 사용하여 얻은 성능 지수로 기존의 모델과 본 논문에서 제시한 유전자 알고리즘을 이용한 SOPNN 모델과의 성능비교를 한 것이다.

### 5. 결론

본 논문에서는 유전자 알고리즘을 이용하여 자기구성 다항식 뉴럴 네트워크의 최적 구조를 설계하였다. 기존의 SOPNN 알고리즘은 층을 구성하는 노드의 입력변수의 수와 다항식 차수 그리고 고정된 입력변수 선택을 설계자의 경험에 의해서 결정함으로써 객관적이고 주어진 데이터의 특성에 맞는 최적의 모델을 구축하지 못하였고, 또한 최적의 모델을 구축하기 위해서 많은 시행착오를 필요로 하였다. 이러한 SOPNN 모델의 구조의 한계를 극복하고자 본 연구에서는 최적화 방법인 유전자 알고리즘을 이용한 각 노드들의 유기적인 조직으로부터 기존의 다른 모델들보다 성능이 우수함을 알 수 있다.

#### 감사의 글

본 연구는 산업자원부의 지원에 의하여 기초천력공학공동연구소(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, W. Pedrycz and B.-J. Park, "Polynomial Neural Networks Architecture: Analysis and Design", Computers and Electrical Engineering, 2002(in press).

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

[5] D. E. Goldberg, Genetic Algorithm in search, Optimization & Machine Learning, Addison wesley, 1989.

[6] S. K. Oh and W. Pedrycz, "Identification of Fuzzy Systems by means of an Auto-Tuning Algorithm and Its Application to Nonlinear Systems", Fuzzy sets and Systems, Vol. 115, No. 2, pp. 205-230, 2000.

[7] S. Horikawa, T. Furuhashi, and Y. Uchigawa, "On Fuzzy Modeling Using Fuzzy Neural Networks with the Back Propagation Algorithm", IEEE trans. Neural Networks, Vol.3, No.5, pp.801-806, 1992.

[8] G. Kang and M. Sugeno, "Fuzzy modeling", Trans. SICE, vol. 23, No. 6, pp.106-108, 1987(in Japanese).

[9] M. Sugeno, Fuzzy Control. Tokyo, Japan:Nikk- an Kogyo Shimbunsha, 1988(in Japanese).

[10] T. Kondo, "Revised GMDH algorithm estimating degree of the complete polynomial", Tran. of the Society of Instrument and Control Engineers, Vol. 22, No. 9, pp. 928-934, 1986.

[11] 오성권, "프로그래밍에 의한 컴퓨터 지능(퍼지, 신경회로망 및 진화알고리즘을 중심으로)", 내하출판사, 2002.