퍼지 및 지능시스템학회 논문지 2005, Vol. 15, No. 2, pp. 236-244

퍼지관계와 유전자 알고리즘에 기반한 진화론적 최적 퍼지다항식 뉴럴네트워크: 해석과 설계

Evolutionally optimized Fuzzy Polynomial Neural Networks Based on Fuzzy Relation and Genetic Algorithms: Analysis and Design

박병준* · 이동윤** · 오성권***

Byoung-Jun Park*, Dong-Yoon Lee**, and Sung-Kwun Oh***

*원광대학교 전기전자 및 정보공학부

**중부대학교 정보통신공학부

***수원대학교 전기공학과

요 약

본 연구에서는 퍼지관계 및 진화론적 최적 다층 퍼셉트론에 기초한 퍼지다항식 뉴럴네트워크(FPNN)의 새로운 구조를 소개하고, 포괄적인 설계방법론을 토의하며, 그리고 일련의 수치적인 실험이 수행된다. 진화론적 최적 FPNN(EFPNN)의 구축을 위해 컴퓨터지능(CI)의 기반 기술을 이용한다. EFPNN의 구조는 규칙베이스 퍼지뉴럴네트워크와 다항식 뉴럴네트워크의 결합에 의한 유전자 최적 구동 하이브리드 시스템의 시너지 이용으로 얻어진다. 퍼지뉴럴네트워크는 EFPNN의 전체 규칙 구조의 전반부에 기여하고, EFPNN의 후반부는 다항식 뉴럴네트워크를 사용하여 설계된다. EFPNN의 후반부를 위한 유전론적 최적 다항식뉴럴네트워크의 개발은 두 최적화 기법에 의해 수행된다. 즉 구조적 최적화는 유전자알고리즘에 의해 수행되고, 파라미터 최적화는 최소자승법 기반의 학습을 통해 행하여진다. EFPNN의 성능 평가를 위해, 모델은 몇 가지 수치 예제를 이용한다. 비교에 의한 해석은 제안된 EFPNN이 이전에 제시된 다른 지능형 모델보다 높은 정확도뿐만 아니라 좀 더 우수한 예측능력을 가지는 모델임을 보여준다.

Abstract

In this study, we introduce a new topology of Fuzzy Polynomial Neural Networks(FPNN) that is based on fuzzy relation and evolutionally optimized Multi-Layer Perceptron, discuss a comprehensive design methodology and carry out a series of numeric experiments. The construction of the evolutionally optimized FPNN(EFPNN) exploits fundamental technologies of Computational Intelligence. The architecture of the resulting EFPNN results from a synergistic usage of the genetic optimization-driven hybrid system generated by combining rule-based Fuzzy Neural Networks(FNN) with polynomial neural networks(PNN). FNN contributes to the formation of the premise part of the overall rule-based structure of the EFPNN. The consequence part of the EFPNN is designed using PNN. As the consequence part of the EFPNN, the development of the genetically optimized PNN(gPNN) dwells on two general optimization mechanism: the structural optimization is realized via GAs whereas in case of the parametric optimization we proceed with a standard least square method-based learning. To evaluate the performance of the EFPNN, the models are experimented with the use of several representative numerical examples. A comparative analysis shows that the proposed EFPNN are models with higher accuracy as well as more superb predictive capability than other intelligent models presented previously.

Key Words : Evolutionally optimized Fuzzy Polynomial Neural Networks (EFPNN), Multi-Layer Perceptron(MLP), Computational Intelligence(CI), Fuzzy Relation-based Fuzzy Neural Networks(FNN), genetically optimized PNN(gPNN)

1. 서 론

많은 분야에서 행해지고 있는 시스템 모델링에 관한 연구의 궁극적인 목표는 자연현상을 이해하고, 분석함으로써 원하는 정보를 이용하는데 있다. 그러나 대부분의 현상들은 근본적으로 비선형을 내포하고 있기 때문에 선형성을 바탕으로 개발된 기존의 방법들을 이용해 이들을 해석하기에는 많은 어려움에 부딪히게 된다. 초기에는 비선형성을 근사적으로 선형화 시킴으로써 이를 극복하려는 시도가 이루어졌으나 근본적인 한계를 넘지는 못하였다. 이후 근본적으로 개념을 달리하는 다양한 시도가 이루어지고 있으며, 퍼지 모델[1,2], 신경회로망[3]과 같은 지능형 이론을 비선형적이고 복잡한 시스템의 특성을 해석하는데 적용함으로써 좋은 결과를 얻게 되었다. 최근에는 CI 기술[4,16]의 발전과 함께 고급 지능형

접수일자 : 2004년 6월 21일
완료일자 : 2005년 2월 14일
감사의 글 : 본 연구는 산업자원부의 지원에 의해 기초전력연구원(R-2004-B-274) 주관으로 수행된 과제임

이론, 즉 퍼지 모델과 신경회로망이 가지고 있는 독특한 특성들을 살리면서 지능 모델의 효율과 성능향상을 위한 두 모델의 융합에 관한 연구가 여러 가지 각도에서 다양하게 제시되고 있다[6-10].

본 논문에서는 CI 기술의 한 영역으로써 효과적인 비선형 문제의 분석과 해결을 위한 고급 CI 기술, 즉 진화론적 최적 퍼지다항식 뉴럴네트워크(Evolutionally optimized Fuzzy Polynomial Neural Networks; EFPNN) 개발을 목표로 하며, 구조 개발을 위한 광범위한 설계방법론을 토의한다. 제안된 EFPNN은 기존에 제시된 퍼지다항식 뉴럴네트워크(FPNN[5,6])와 진화 연산인 유전자 알고리즘(Genetic Algorithms; GAs[11,12])과의 융합에 의해 생성되는 최적 구조이다. EFPNN은 FPNN 모델과 마찬가지로 전반부와 후반부 구조로 구성된다. EFPNN의 전반부 구조는 간략 및 선형 퍼지추론과 신경망의 융합에 의해 설계된 퍼지뉴럴네트워크(Fuzzy Neural Networks; FNN[5,6])에 의해 설계되며, 하이브리드 학습방법 즉, GAs와 오류역전파(Back-propagation; BP) 학습 알고리즘을 이용하여 파라미터를 최적화한다. EFPNN의 후반부 구조는 FPNN 후반부 구조와는 달리 GAs와 다항식 뉴럴네크워크(Polynomial Neural Networks; PNN[7,8])의 융합에 의한 구조적 최적화로부터 구현된다. 이를 위해 유전론적 최적 PNN(gPNN)을 설계한다. gPNN의 구조는 자기구성과 적자생존의 자연법칙에 기초를 둔 PNN과 GAs의 상호 융합을 통해 행해지며, 파라미터는 최소자승법에 의해 최적화된다. 기존의 FPNN의 후반부 구조는 고정된 입력 수와 다항식 차수를 이용하여 네트워크의 성장과정을 통해 다양한 구조로 설계되지만, 최적 네트워크 구조임을 보증하지 못하다. 그러나 제안된 EFPNN은 GAs에 의해 구조적으로 최적화된 네트워크가 되도록 설계되며, 기존의 모델링 방법들 보다 훨씬 유연성과 정확성을 가진 적응구조를 생성한다. 제안된 EFPNN 구조의 평가를 위해 몇 개의 비선형 공정을 이용한다. 주어진 공정에 제안된 구조를 적용하고 이에 대한 결과 분석 및 평가를 통해 구조의 정확성, 타당성 등을 검증한다.

2. 기존의 퍼지다항식 뉴럴네트워크 구조

하이브리드 구조 형태인 자기구성 퍼지다항식 뉴럴네트워크(FPNN[5,6])는 FNN과 PNN의 상호 결합된 구조로 FNN은 FPNN 구조의 전반부를, PNN은 후반부를 구성한다. FPNN의 전반부 구조인 FNN은 간략 퍼지추론, 선형 퍼지추론, 오류역전파(BP) 학습 규칙을 이용하며, 각 구조의 멤버쉽함수 파라미터는 GAs에 의해 최적화된다. 후반부 구조로 이용된 PNN은 수학적인 다항식 형태로 표현되며 GMDH(Group Method of Data Handling [13])에 기반을 둔다. GMDH는 입력변수의 선택과 2변수 2차식의 부분표현식에 의해 시스템 방정식을 예측하였다. 그러나 2변수의 입력변수는 입력변수들 사이의 상호의존적인 면이 약하고 또한 2차식으로 모든 경우를 표현함으로써 해당 공정모델의 적응력을 감소시켰다. PNN은 이러한 문제를 해결하였으며, 학습을 통해 발전하는 유연한 구조로 층과 다항식 뉴런(Polynomial Neuron; PN)의 생성 및 소멸을 통해 구축된 네트워크이다. FPNN은 FNN과 PNN의 장점을 가지며 상호 보완의 특성을 가진다. 전반부 FNN 사용은 구조의 특성상 4 입력 이상의 시스템 입력변수를 요구하는 PNN의 단점을 보완함으로써 적은 입력변수를 가진 시스템에 대해서도 PNN의 다양한 구조 적용을 용이하게 하며, 후반부에 PNN을 사용함으로써 FNN의 학습방법인 오류역전파 학습방법을 도와 좀더 향상된 성능을 나타낸다.

FPNN의 구조는 전반부와 후반부의 구조에 따라 다양하게 설계된다[5,6]. 그림 1은 전반부와 후반부 구조에 따라 설계되는 FPNN의 구조를 나타낸다. FPNN의 구조는 주어진 공정의 특성에 따른 전반부의 구조와 후반부 구조의 선택에 의해 적응력을 가지는 구조로 설계된다. 특히, 후반부 구조인 PNN의 설계에 따라 FPNN의 구조는 결정된다.

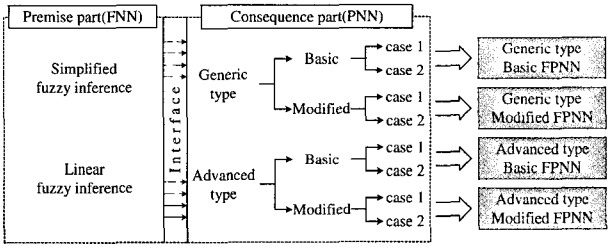


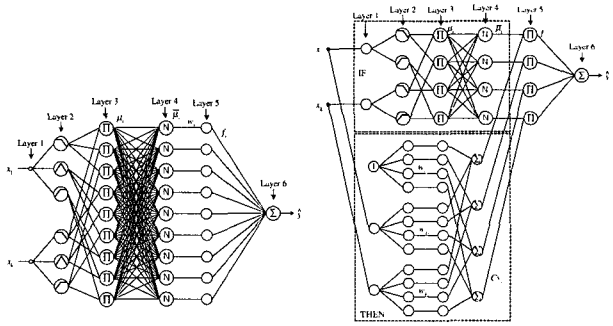
그림 1. 기존의 FPNN 구조
Fig. 1. Family of the conventional FPNN

3. EFPNN의 구조와 개발

본 논문에서 제안된 진화론적 최적 퍼지다항식 뉴럴네트워크(EFPNN) 구조는 진화에 의해 생성된다. 전반부 구조는 FNN의 수행단계 따라 후반부로 전달한다. 전달된 신호는 후반부의 입력변수들이 되며, 이들 입력변수들과 GAs에 의해 최적 PN을 선택하는 진화과정을 수행하게 된다.

3.1 퍼지뉴럴네트워크와 GAs에 의한 최적화

EFPNN의 전반부에 이용되는 퍼지뉴럴네트워크(FNN)는 비선형적인 입력 데이터들간의 상호 연관성을 찾아 이들 사이에 정의된 퍼지관계를 이용하여 공간을 분할한다. 모든 변수들의 관계를 동시에 고려한 공간분할을 통해 형성된 퍼지언어 정보로부터 핵심 역할을 수행한다. 관련 FNN 구조는 그림 2와 같다.



(a) Simplified FNN (b) Linear FNN

그림 2. 입력변수들의 퍼지관계에 기반한 FNN 구조
Fig. 2. Topology of FNN based on fuzzy relation for the input variables

그림 2에서 '원'은 FNN의 뉴런을 의미하며, 'Π' 뉴런은 들어오는 신호를 모두 곱함으로 출력을 생성한다. 'N' 뉴런은 신호들의 정규화 과정, 'Σ' 뉴런은 선형 합을 의미한다.

FNN 구조는 표 1에서 보여지는 것과 같은 퍼지규칙으로 이루어진다. R^i는 i번째 퍼지규칙을 나타내며, A_{ki}는 퍼지규칙의 전반부 멤버쉽함수로 보수적인 관계를 가진 삼각형 형태이다. w_i와 w_{ki}는 상수로, 그림 2의 연결 가중치를 나타낸다. 그림 2와 표 1에서 보여지는 것과 같이 FNN은 퍼지규칙 후반부의 형태에 따라 설계된다. 간략추론 구조는 연결가중치 w_i가 퍼지규칙 후반부의 상수항을 나타내며, 반면 선형추론 구조를 가지는 FNN은 연결가중치에 의한 네트워크가 선형식을 표현하여 규칙 후반부를 선형식으로 나타내며, 선형 퍼지추론에 의해 모델의 출력을 구한다.

표 1. 간략 및 선형 퍼지추론 기반 FNN의 비교
Table 1. Comparison of simplified with linear fuzzy inference-based FNNs

FNN 구조		간략 추론구조	선형 추론구조
퍼지 규칙	전반부	R^i : *If* x_1 *is* A_{1i} and x_k *is* A_{ki}	
	후반부	*then* $Cy_i = w_i$	*then* $Cy_i = w_{0i} + w_{1i} \cdot x_1 + w_{ki} \cdot x_k$
추론결과		$\hat{y} = \sum_{i=1}^{n} f_i = \sum_{i=1}^{n} \overline{\mu_i} \cdot Cy = \sum_{i=1}^{n} \frac{\mu_i \cdot Cy_i}{\sum_{i=1}^{n} \mu_i}$	
학습		$\Delta w_i = 2 \cdot \eta \cdot (y - \hat{y}) \cdot \overline{\mu_i} + \alpha(w_i(t) - w_i(t-1))$	$\Delta w_{0i} = 2 \cdot \eta \cdot (y - \hat{y}) \cdot \overline{\mu_i} + \alpha(w_{0i}(t) - w_{0i}(t-1))$ $\Delta w_{ki} = 2 \cdot \eta \cdot (y - \hat{y}) \cdot \overline{\mu_i} \cdot x_k + \alpha(w_{ki}(t) - w_{ki}(t-1))$

FNN의 퍼지추론 과정은 각 퍼지규칙의 적합도를 구하고 정규화된 적합도를 계산한다. 정규화된 적합도 $\overline{\mu_i}$는 연결가중치와 곱하며, 출력층의 입력이 된다. f_i 뉴런의 출력은 그림 2와 표 1에서 보여지는 것과 같이 모두 대수적으로 합해지며, 무게중심법에 의해 $\hat{y}$을 최종적으로 출력하게 된다.

FNN의 학습에는 오류역전파(BP) 알고리즘이 사용되며, 그림 2의 연결가중치 w_i와 w_{ki}를 조절함으로써 이루어진다. 모멘텀을 포함한 연결가중치의 변화량 Δw를 표 1에 보였다. 여기서 y는 주어진 데이터, $\hat{y}$는 모델의 출력, η는 학습율, α는 모멘텀계수이며 0과 1 사이의 값을 갖는다. $w_i(t)$는 현재 연결가중치, $w_i(t-1)$은 이전 연결가중치이다. EFPNN의 전반부에 이용되는 FNN의 최적화는 하이브리드 방법에 의한다. 즉, 멤버쉽함수의 파라미터, 학습율, 모멘텀계수는 GAs에 의해 최적화되며, 동시에 BP에 의해 연결가중치가 조절된다..

3.2 유전자 기반 다항식 뉴럴네트워크

본 논문에서 제안한 EFPNN의 최적 구조를 위해 유전자 기반 다항식 뉴럴네트워크(gPNN)을 설계한다. gPNN은 최적화 탐색 방법인 GAs을 이용하여 PNN의 입력변수 수와 이에 해당되는 입력변수 그리고 다항식의 차수를 동조함으로써 생성된다. 즉, 최적의 PNN 구조이다. 기존의 PNN은 네트워크의 성장과정을 통하여 각층의 다항식 뉴런(PN)에서 고정된(설계자에 의해 미리 선택된) 노드 입력의 수뿐만 아니라 다항식 차수(1차, 2차, 변형된 2차)를 이용하였다. 이러한 PNN은 학습을 통해 발전하는 구조로 층과 PN 수가 고정되어 있지 않고 생성 및 소멸하면서 네트워크를 구성한다. PNN 구조는 주어진 비선형 문제에 높은 유연성을 보여주며, 공정 모델 특성과 계층구조의 간략성 및 상호 연계성 고려로부터 고차 회귀다항식 사용이 용이하고 간단한 구조로 설계된다[7,8]. 그러나 PNN 구조는 한 층에서의 PN 모두가 똑같은 입력변수의 수와 다항식 차수를 가지기 때문에 모델 구축에 있어서 다양성과 데이터 특성에 맞는 모델을 구축하지 못하는 단점이 있었다. 즉, 하나의 층이 그림 1에서 보여준 것과 같이 Basic, Modified, case 1, case 2와 같이 구분되어 설계되었다. 따라서 PN의 입력 수와 변수의 선택 및 다항식의 차수 결정 등을 해결해야하는 문제를 가지고 있다. 이들 문제들은 주어지는 공정에 대해 다르게 나타나고 모델 성능에 영향을 미치기 때문에 전문가의 지식을 요구하게 된다. 이처럼 각 층의 입력변수, 차수 등을 설계자가 경험에 의해서 결정해야함으로써 최적의 네트워크를 구하는데 있어서 다양한 네트워크 구조와 그와 관련된 다양한 방법론적 설계가 요구된다. 본 논문에서는 PNN 구조가 가지고 있던 단점을 해결하고, 전문가의 경험에 의존한 구조의 최적화를 객관적이고 효율적으로 구축하기 위해 GAs를 적용한다. GAs와 PNN의 융합은 다양한 구조와 방법론적 설계를 요구하지 않으며 단일화된 최적 네트워크 구조인 gPNN을 생성한다. 그림 3은 GAs에 의한 PNN 각 층의 최적화 과정을 보여주고 있다.

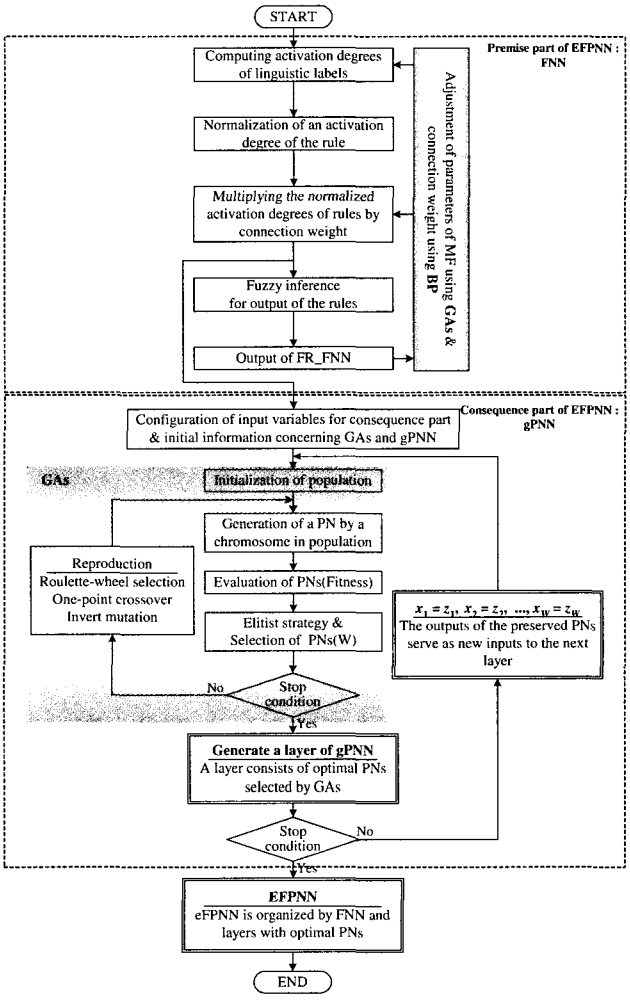


그림 3. EFPNN 구조를 위한 전체 설계 흐름도
Fig. 3. Overall design flowchart for EFPNN architecture

4. EFPNN의 알고리즘과 설계과정

EFPNN은 GAs를 이용하여 기존의 FPNN을 최적화한 구조로 진화하며, 네트워크를 생성해 간다. EFPNN 구조는 전반부와 후반부로 구성된다. 전반부 구조인 FNN과 후반부를 구성하는 gPNN은 각각 GAs에 의해 진화하여 최적 구조로 설계된다.

4.1 EFPNN의 전반부; FNN

[Layer 1] 입력층; 각 입력신호를 다음 층으로 보낸다.
[Layer 2] 멤버쉽함수에 대한 활성값 계산; 이 층의 각 뉴런은 입력변수에 대해 정의된 퍼지집합에 의해 멤버쉽함수의 활성값이 결정된다.
[Layer 3] 퍼지규칙의 전반부 적합도 계산; 각각의 뉴런은 'Π'에 의해 정의되며, 퍼지규칙의 적합도가 된다. 이 층의 각 뉴런은 하나의 퍼지규칙을 나타내며, 출력은 각 규칙의 전반부 적합도이다.
[Layer 4] 정규화된 전반부 적합도 계산; 각 규칙에 대한 적합도가 구해지면, 정규화된 적합도 $\overline{\mu}_i$를 계산한다.

$$\overline{\mu}_i = \frac{\mu_i}{\sum_{i=1}^{n} \mu_i} \tag{1}$$

여기서 n은 퍼지규칙의 수이다.
[Layer 5] 연결 가중치와의 곱; 4층에서 구해진 적합도는 연결가중치에 의한 후반부 출력과 곱해져 각 퍼지규칙의 출력을 생성하고 다음 층의 입력이 된다.

$$f_i = \overline{\mu}_i \times Cy_i \tag{2}$$

연결가중치에 의한 퍼지규칙의 후반부 출력 Cy_i는 퍼지추론 구조에 따라 다음과 같이 표현된다.

$$\begin{cases} \text{Simplified} : & Cy_i = w_i \\ \text{Linear} : & Cy_i = w_{0i} + w_{1i} \cdot x_1 + w_{ki} \cdot x_k \end{cases} \tag{3}$$

EFPNN의 전반부인 FNN은 후반부 구조인 gPNN과 5층에서 연결된다. 즉, 각 퍼지규칙의 출력 f_i는 gPNN의 입력이 된다.
[Layer 6] FNN의 출력 : 5층의 출력 f_i는 표 1의 추론 결과와 같이 모두 대수적으로 합해져 y을 최종적으로 출력하게 된다.

4.1 EFPNN의 후반부; gPNN

[단계 1] 입력변수의 설정 : EFPNN의 전반부 구조 5층 출력 f_i는 후반부 구조인 gPNN의 입력변수가 된다. 즉, $x_1=f_1$, $x_2=f_2, \cdots, x_n=f_n$가 된다.
[단계 2] 기본 정보 결정 : EFPNN 구조를 생성하기 위해 gPNN의 최대 층 수, PN에 입력될 최대 입력변수의 수, 하나의 층에서 생성될 PN의 수 등을 정의한다.
[단계 3] 초기 세대 생성 : 초기 세대를 생성하고, 생성된 염색체로부터 개체 수만큼의 PN을 구현한다. 여기서 하나의 개체는 gPNN을 구성하는 하나의 PN 정보를 가지게 된다.
[단계 4] GAs의 염색체 정보에 의한 PN 결정 : 염색체에 의한 PN 구조의 결정은 그림 4와 같다.
[단계 4-1] 입력변수의 수 선택
Step 1) 주어진 전체 염색체 정보 중 처음 3비트는 입력변수의 수를 결정하기 위해 이용된다. 설계자에 의해 이 비트 수는 달라질 수 있다.
Step 2) 선택된 3비트를 10진수로 복호화 한다.
Step 3) 복호화된 10진수를 설계자가 정한 최대 입력변수의 수(N)로 정규화한다.
Step 4) Step 3에서 정의된 정수(NI)는 하나의 PN를 구축하는 입력변수의 수로 선택되어진다.
[단계 4-2] 다항식 차수 선택
Step 1) 입력변수 수의 선택 비트 이후의 3비트를 다항식 차수 선택을 위한 비트로 설정한다. 이 비트 수 역시 설계자에 의해 달라질 수 있다.
Step 2) 선택되어진 3비트를 10진수로 복호화 한다.
Step 3) 10진수를 1~3 사이의 정수로 정규화 한다.
Step 4) 결정된 값은 PN의 다항식 차수로 설정된다[7,8].
- 1의 선택 → Type 1 : 선형식
- 2의 선택 → Type 2 : 2차 다항식
- 3의 선택 → Type 3 : 변형된 2차 다항식

[단계 4-3] 입력변수 선택
Step 1) 염색체 정보의 나머지 비트는 입력변수 선택을 위해 사용된다.

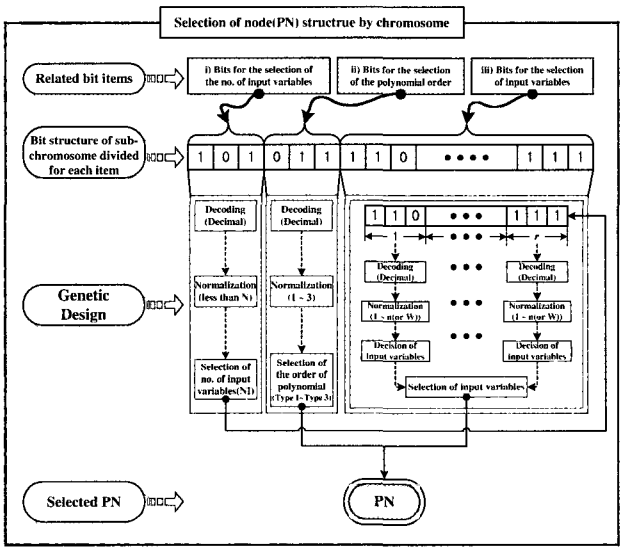


그림 4. GAs의 염색체를 이용한 PN 설계
Fig. 4. PN design by chromosome of GAs

Step 2) 입력변수 선택 비트를 [단계 4-1]에서 구한 입력변수의 수(NI)로 균등 분할한다.
Step 3) 분할된 각 비트들을 10진수로 복호화 한다.
Step 4) 복호화된 값은 1에서 전체 입력변수의 수 사이의 정수로 정규화 한다.
Step 5) 정규화된 값들은 입력변수의 번호가 된다. 즉 PN의 입력변수가 된다. 만약 선택된 입력변수가 중복되면, 하나의 변수로 처리한다.
[단계 5] PN의 매개변수 추정 및 평가 : 단계 4에서 결정된 정보로부터 PN의 매개변수(파라미터)를 추정하고 평가한다. 매개변수의 추정은 선형 회귀분석에 의해 학습용 데이터를 사용하여 추정한다[7,8]. 추정된 PN의 근사화와 일반화 능력을 평가하기 위해 식 (4)과 같은 적합도 함수를 이용한다. 목적함수(Objective function)는 다음 장에서 각 공정에 대해 제시하였다.

$$F = \frac{1}{1 + Objective\ function} \tag{4}$$

[단계 6] PN 선택과 엘리트 전략 : 구해진 적합도(F_1, F_2, …, F_z)를 선호되는 적합도 순으로 정렬한다. 정렬된 적합도에 따라 EFPNN 후반부 구조의 생성을 위해 중복된 개체를 제거하고 W개의 개체를 선택한다. 초기 세대 이후의 세대는 다시 이전 세대 개체와의 비교를 통해 중복된 개체를 제거하고 두 세대의 최적 개체 W개를 선택한다. 여기서 W는 EFPNN의 후반부 각 층 구조를 위해 선택되는 PN의 수이다. 선택된 개체들 중 최적의 적합도를 가지는 개체를 선택하여 엘리트 전략을 수행한다.

[단계 7] 재생산 : 다음 세대의 생성을 위해서 염색체 정보와 개체의 적합도 등을 이용하여 선택, 교배, 돌연변이 연산을 수행한다. 마지막 세대까지, 단계 4부터 단계 7을 반복한다.

[단계 8] EFPNN의 후반부 층 결정 : GAs에 의해 마지막 세대까지 진화한 개체들은 W개의 최적 PN을 생성하며, 생성된 PN들은 EFPNN의 후반부를 구성하는 하나의 층을 이루게 된다.

[단계 9] 종료판정 : 종료판정은 네트워크의 복잡성과 우수성 등의 상호균형을 고려하여 알고리즘을 종료한다.

[단계 10] 다음 층의 새로운 입력변수 설정 : GAs에 의해 생성된 현재 층의 PN 출력들(z_1, z_2, …, z_W)로부터 다음 층의 새로운 입력들(x_1, x_2, …, x_W)을 $x_1=z_1$, $x_2=z_2$, …, $x_W=z_W$으로 구성하고, [단계 3]로 간다. 이 후, [단계 3]부터 [단계 10]까지를 반복 수행한다.

[단계 11] 종료 : 알고리즘이 종료하면 마지막 층의 최적 PN의 다항식에 전 층의 PN을 대입하고 동일한 조작을 1층까지 반복하여 최종 추정 모델 $\hat{y}$를 얻는다.

FNN과 gPNN의 합성 구조인 EFPNN 구조는 주어진 비선형 문제를 해결하기 위해 구조 설계와 구조의 결정, 방법론적 설계 등의 과정이 생략된다. 단지 GAs의 염색체 정보를 가지고 문제해결의 최적 해를 찾아 진화하며, 최적의 구조를 구축하게 된다. 따라서, 전문가의 지식을 필요로 하는 많은 부분을 배제함으로써 객관적이고, 유연성, 정확성, 예측능력을 가진 모델을 설계할 수 있다.

5. 실험연구 및 결과고찰

본 논문에서는 제안된 EFPNN 구조의 평가와 분석을 위해 두 종류의 데이터를 이용한다. 첫 번째는 비선형 함수[1,10,17]로 3입력 1출력의 구조를 가지는 수치 예제이다. 두 번째는 NOx 배출공정[18]에 모델을 적용한다. 각 공정에 적용된 EFPNN의 구조 생성을 위해 공통적으로 적용되는 GAs의 연산자 중 선택(Selection) 연산자는 룰렛-휠(Roulette-wheel) 선택, 교배(Crossover) 연산자는 일점(One-point) 교배, 돌연변이(Mutation) 연산자는 반전(invert) 돌연변이를 사용한다. 교배율과 돌연변이율은 각각 0.75와 0.065를 적용하였다. 그 외 GAs의 파라미터는 각 실험연구에서 제시한다. 각 예제의 성능지수(목적함수)는 비선형 함수에 대해 식 (5)를, NOx 배출 공정에 대해 식 (6)을 이용한다.

$$PI \text{ or } E_PI = \frac{1}{n}\sum_{p=1}^{n}\frac{|y_p-\hat{y}_p|}{y_p}\times 100(\%) \tag{5}$$

$$PI \text{ or } E_PI = \frac{1}{n}\sum_{p=1}^{n}(y_p-\hat{y}_p)^2 \tag{6}$$

5.1 비선형 함수

제안된 모델의 적용 및 유용성을 비교 평가하기 위하여 식 (7)로 표현되는 비선형 함수를 이용한다. 이 함수는 Sugeno[1,10]가 제안한 퍼지추론 및 파라미터 동정 기법에 적용된 함수로 Sugeno가 사용한 40개의 입출력 데이터를 본 연구에 적용함으로써 제안된 각각의 모델을 비교하며, 타당성 및 정확도를 검토한다.

$$y=(1.0+x_1^{0.5}+x_2^{-1}+x_3^{-1.5})^2 \tag{7}$$

EFPNN의 비선형 함수 적용을 위한 설계정보와 GAs의 염색체 정보는 표 2와 같다. 유전자 정보 중 엘리트 군집 크기(Elite population size)는 선택되어질 최적 PN의 최대수(W)로, EFPNN의 후반부 한 층을 구성하며, 이들의 출력은 다음 층의 입력변수들이 된다. 따라서 후반부 1층에서의 입력 수는 8이지만 2층 이상에서는 선택된 PN 수만큼의 입력변수를 가지게 된다. 문자열 길이(String length)는 (3+3+28)로, (입력변수의 수 선택비트+다항식 차수 선택 비트+입력변수 선택비트)를 의미한다. 즉, PN의 입력변수 수를 결정하는데 3비트, PN의 다항식 차수를 결정하는데 3비트, PN의 입력변수를 선택하는데 28비트를 사용한다. 문자열의 길이는 적절히 선택하여야 한다. 적은 수의 문자열은 원하는 정보를 주지 못하며, 많은 수의 문자열은 느린 탐색과 많은 시간을 요구하고, 최적의 모델을 구축하지 못하는 경우가 발생할 수 있다. EFPNN의 설계정보는 전반부와 후반부로 나뉘어 각각 설정된다. 전반부의 입력 수는 주어진 시스템 입력변수의 수이며, EFPNN의 입력이 된다. 전반부 구조의 학습율 및 모멘텀계수는 GAs에 의해 동정된 값이다. 전반부 구조인 FNN은 각 입력변수에 대해 2개의 멤버쉽함수를 적용함으로써 8개의 퍼지규칙으로 구성된다. 이들 퍼지규칙의 출력은 후반부 구조와 연결이 되며, gPNN의 입력이 된다. 여기서 전반부 구조인 FNN은 그림 2의 5층에서 연결점을 가진다. 이 연결점으로부터 후반부 구조는 8개의 입력변수를 가진다. 후반부 gPNN은 각 전반부 구조에 대해 주어진 입력변수 중 4개보다 적은 입력을 선택($1\le N\le 4$)하여 GAs에 의해 PN을 구성하고, 엘리트 집단(W)을 선택하여 gFPNN 후반부의 한 층을 이룬다. 후반부 층을 형성하는 선택된 PN들로부터 입력변수를 다시 형성하고, 다음 층을 생성함으로써 네트워크 구조를 구축해 나간다.

표 2에 의한 EFPNN 구조의 결과는 표 3과 같다. 여기서 'FI'는 퍼지추론, 'NR'은 퍼지규칙 수', MF'는 멤버쉽함수, 'L'은 층, 'NI'는 선택된 PN의 입력변수 수, 'Input No.'는 각 층에서 최적으로 선택된 PN의 입력변수, 'T'는 후반부 구조를 구성하고 있는 PN의 다항식 차수, PI는 학습데이터에 대한 성능지수, E_PI는 테스트 데이터에 대한 성능지수를 나타낸다. 표 3의 결과는 FNN과 gPNN의 결합에 의한 EFPNN 후반부 각 층에 대한 최적 PN의 성능지수를 나타낸다. 예를 들어 선형 퍼지추론 구조의 FNN과 gPNN의 연결구조를 보면, EFPNN의 후반부 1층에서 최적 PN은 x_1, x_5, x_6, x_7(전반부 구조의 측면에서 보면 f_1, f_5, f_6, f_7이 된다.)의 입력변수들이 선택되었고, 다항식은 2차식으로 구성된다. 이때 성능지수는 PI=0.423, E_PI=4.601이다. 입력변수의 경우, 후반부 구조의 1층은 전반부와의 연결에 의한 입력이 되지만, 2층 이상은 전 층의 PN 출력이 된다. 따라서 2층 이상의 입력변수 번호는 전 층을 구성하고 있는 PN의 번호가 된다. 즉, 2층의 최적 PN은 1층을 구성하는 PN들 중 3, 34, 44 그리고 49번째 PN의 출력으로부터 4입력을 구성하고, 3차식의 다항식으로 표현된다. 이때 PI=0.184, E_PI=2.175의 출력특성을 가진다.

3층, 4층, 5층도 마찬가지로 구성된다.

표 2. GAs와 EFPNN의 관련 정보
Table 2. Information related to GAs and EFPNN

GAs	Generation			150
	Population size			100
	Elite population size(W)			50
	String length			3+3+28
EFPNN	Premise (FNN)	No. of entire inputs		3
		Iteration		1000
		Learning rate	Simplified	0.309
			Linear	0.879
		Momentum coefficient	Simplified	0.056
			Linear	0.022
		No. of rules		8
	Consequence (gPNN)	No. of entire inputs		8
		Layer		5
		No. of inputs to be selected(N)		$1 \le N \le 4$
		Type(T)		$1 \le T \le 3$

N, T : integer

표 3. 비선형 함수에 대한 EFPNN의 성능지수
Table 3. Performance index of EFPNN for the nonlinear function

Premise part		Consequence part							PI	E_PI
FI	NR(MFs)	L	NI	Input No.				T		
Simplified	8 (2×2×2)	1	4	5	6	4	7	1	8.138	10.68
		2	4	2	45	49	25	2	0.382	2.316
		3	4	42	9	40	49	3	0.313	1.403
		4	3	36	16	33	·	1	0.311	0.734
		5	3	43	45	28	·	2	0.309	0.610
Linear	8 (2×2×2)	1	4	6	1	5	7	2	0.423	4.601
		2	4	49	4	44	34	3	0.184	2.175
		3	2	43	2	·	·	2	0.105	1.361
		4	3	33	44	5	·	3	0.063	0.761
		5	3	41	26	42	·	3	0.039	0.587

전반부 구조는 앞에서 언급한 것과 같이 FNN의 간략 및 선형 퍼지추론 구조를 이용하였고, 각 입력변수에 대해 2개의 멤버쉽함수를 적용하여 $8(2^3)$개의 규칙을 가진다. 이때 간략 퍼지추론 구조의 FNN 성능지수는 PI=3.997, E_PI=3.269, 선형 퍼지추론 구조는 PI=2.069, E_PI=2.518이다. 제안된 EFPNN의 구조는 결과에서 보여지는 것과 같이 FNN의 성능을 개선하였고 전반부에 선형 퍼지추론 구조를 가지는 모델에서 좋은 출력을 얻을 수 있었다. 또한 후반부 층이 증가할수록 모델의 근사화와 일반화 능력이 향상되고 있음을 알 수 있다. 따라서 구조의 복잡성과 모델의 성능 등을 고려하여 적절한 층을 선택하도록 한다.

그림 5는 본 논문에서 제안된 방법론에 의해 설계되어진 EFPNN 구조를 보여주고 있다. 이 것은 선형 퍼지추론 구조의 FNN과 gPNN의 합성 구조이며, 이들 구조의 각 층 성능지수는 표 3에 보여진다. 그림에서 각 층을 이루는 PN은 다음과 같이 표현된다.

: PN_n - 각 층을 구성하고 있는 n번째 PN
NI - PN_n으로 들어오는 입력 수
T - PN_n의 다항식 차수

제안된 EFPNN 구조는 복잡성과 모델의 성능 등을 고려하여 적절한 층을 선택할 수 있다. 또한 모델을 구축하기 전, 후반부 PN의 구조를 결정해야하는 문제를 해결하였고, 설계자에 의해 설정된 일정한 형식 없이 객관적이고 효율적으로 최적의 모델을 제공한다. EFPNN의 전반부 및 후반부의 최적화 과정은 그림 6과 같다. 여기서 전반부는 BP에 의한 학습과정을, 후반부는 진화에 의한 각 층의 성능지수를 나타낸다.

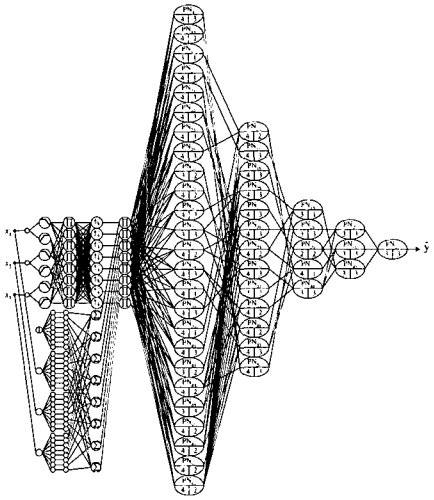


그림 5. 비선형 함수에 대한 EFPNN의 최적 구조
Fig. 5. Optimal topology of EFPNN for the nonlinear function

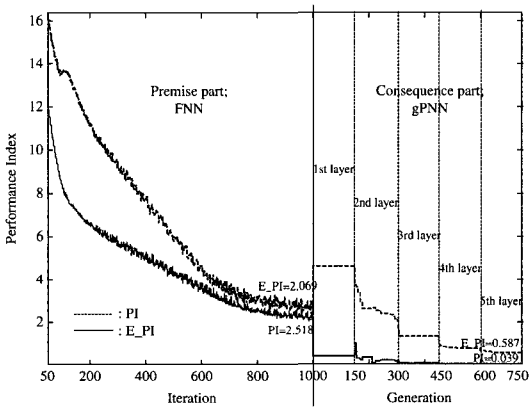


그림 6. GAs와 BP에 의한 EFPNN의 최적화 과정
Fig. 6. Optimization procedure of EFPNN by GAs and BP

표 4는 3입력 1출력의 비선형 함수에 적용한 기존의 모델과 본 논문에서 제시한 모델과의 성능을 비교하여 보여준다. 여기서 Sugeno의 퍼지모델 I, II는 다차원 추론방식에 기초한 퍼지모델이고 Shin-ichi의 모델은 신경회로망의 학습기능

을 이용하여 퍼지규칙을 동정한 방법이다. Multi-FNN은 클러스터를 이용한 구조로 3개의 FNN으로 구성된다. FPNN의 MFPNN 모델은 전반부가 8개의 퍼지규칙으로 구성되며, 후반부를 구성하는 다항식이 4 입력 Type 3로 고정되어있다. BFPNN은 전반부가 퍼지집합에 의해 구현되며, 후반부는 4 입력 Type 2로 구성된다. 결과에서 보여지는 것처럼 제안된 구조들에서 좋은 출력특성을 얻을 수 있으며, 근사화 및 일반화 능력이 우수함을 알 수 있다.

표 4. 기존 모델방법과의 동정오차 비교
Table 4. Comparison of performance with other modeling methods

Model		PI	E_PI	No. of rules
Linear model[17]		12.7	11.1	
GMDH[14,17]		4.7	5.7	
Sugeno's[1,17]	Fuzzy모델 I	1.5	2.1	3
	Fuzzy모델 II	1.1	3.6	4
Shin-ichi's[10]	FNN Type 1	0.84	1.22	$8(2^3)$
	FNN Type 2	0.73	1.28	$4(2^2)$
	FNN Type 3	0.63	1.25	$8(2^3)$
FNN[15]	Simplified	2.865	3.206	9(3+3+3)
	Linear	2.670	3.063	9(3+3+3)
Multi-FNN[15]	Simplified	0.865	0.956	9(3+3+3)
	Linear	0.174	0.689	9(3+3+3)
FPNN[5]	BFPNN	0.299	0.555	6 rules/5th layer
	MFPNN	0.116	0.360	8 rules/5th layer
The proposed EFPNN	Simplified	0.309	0.610	8 rules/5th layer
	Linear	0.039	0.587	8 rules/5th layer

5.2 가스 터빈 발전소의 NOx 배출공정

대기상 질소산화물은 주로 연소공정을 통해 배출되며 산업공정, 발전소, 자동차 배기가스 그리고 난방에 의해서 발생된다. 그 중 대기에서 문제가 될 만큼 존재하는 것들은 N2O, NO 및 NO2이고 이들 물질을 대기오염 측면에서 NOx(질소산화물)로 일컫는다. NOx의 저감방법으로는 발생원으로부터 NOx의 배출을 억제하는 방법과 배출가스중의 NOx를 제거하는 방법을 들 수 있으며, 최근에는 Gas Turbine에 사용하는 주변 환경변수를 입력변수로, 출력변수는 NOx양으로 배출 메카니즘을 모델링하고, 최적 모델을 구축하여 시뮬레이터를 구성함으로써 NOx를 효과적으로 관리하는 방법이 연구되고 있다.

본 논문에서는 이러한 입출력 변수들을 이용하여 제안된 모델의 성능을 평가한다. 평가를 위해 전체 260개의 데이터를 랜덤하게 130개씩 선택하여 학습데이터와 테스트 데이터로 사용하며, 모델의 근사화와 일반화 성능을 얻는다. 근사화와 일반화 능력을 측정하는 성능지수는 식 (6)과 같다.

FNN과 gPNN의 결합구조인 EFPNN 구조의 설계를 위한 정보와 GAs의 염색체 정보는 표 5와 같다. EFPNN의 전반부 구조인 FNN은 퍼지규칙의 수가 32로 이것은 후반부 구조의 입력 수가된다. 따라서 후반부를 구성하는 PN의 입력변수 수로 $1 \le N \le 32$를 선택하게되면, 과도한 연산시간 및 파라미터의 증가를 발생시킨다. 반대로 적은 수를 선택하게 되면, 입출력에 대한 전반부와 후반부 구조와의 상호관계를 고려하지 못하고 많은 정보를 버리기 때문에 모델의 성능을 저하시키는 결과를 가져온다. 본 논문에서는 이러한 문제를 고려하여 $1 \le N \le 15$를 선택하도록 하였다. 이를 바탕으로 EFPNN은 표 6과 같은 결과를 가진다.

전반부 구조인 FNN은 앞의 예제와 마찬가지로, 간략 및 선형 퍼지추론 구조를 가지며, GAs에 의해 파라미터를 동조하였다. 이때 간략 퍼지추론 구조는 PI=0.711, E_PI =1.699, 선형추론 구조는 PI=0.079, E_PI=0.204로 32개의 퍼지규칙을 가진다. 후반부는 32개의 퍼지규칙의 출력을 입력으로 받아 EFPNN을 수행한다. 제안된 EFPNN은 근사화 및 일반화 성능을 고려하였을 때 간략 퍼지추론 구조에서 좀더 나은 결과를 보여준다. 그림 7(a)는 학습 데이터에 대한 EFPNN의 출력과 원 출력을 비교하여 보여주며 그때 모델오차는 그림 7(b)와 같다. 그림 8은 테스터 데이터에 대한 모델출력과 오차이다. 그림에서 보여지는 것과 같이 모델 출력은 전체(학습 및 테스트) 데이터에 대해 ±1.의 오차를 가진다.

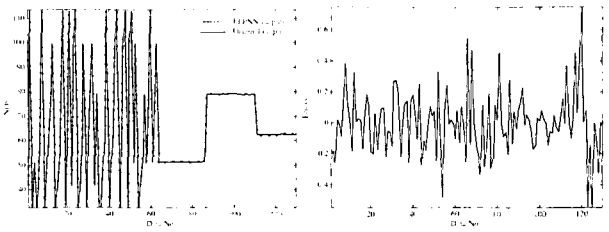
표 5. GAs와 EFPNN의 관련 정보
Table 5. Information related to GAs and EFPNN

GAs	Generation			150
	Population size			100
	Elite population size			50
	String length			3+3+105
EFPNN	Premise (FNN)	No. of entire inputs		5
		Iteration		1000
		Learning rate	Simplified	0.568
			Linear	0.651
		Momentum coefficient	Simplified	0.044
			Linear	0.064
		No. of rules		32
	Consequence (gPNN)	No. of entire inputs		32
		Layer		5
		No. of inputs to be selected(N)		$1 \le N \le 15$
		Type(T)		$1 \le T \le 3$

N, T : integer

표 6. NOx 배출공정에 대한 EFPNN의 성능지수
Table 6. Performance index of EFPNN for the NOx emission process

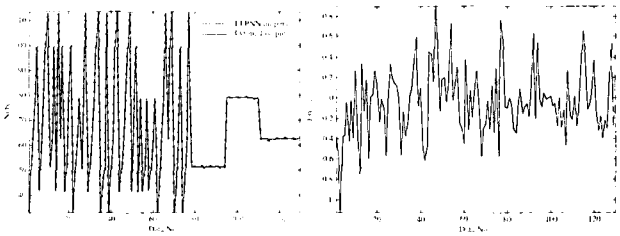
Premise part		Consequence part			PI	E_PI
FI	NR(MFs)	L	NI	T		
Simplified	32 (2×2×2×2×2)	1	13	2	0.149	0.921
		2	12	1	0.065	0.189
		3	11	1	0.046	0.134
		4	8	1	0.044	0.125
		5	3	3	0.041	0.111
Linear	32 (2×2×2×2×2)	1	10	2	0.205	1.522
		2	13	1	0.049	0.646
		3	3	3	0.028	0.437
		4	12	1	0.023	0.330
		5	3	3	0.019	0.286



(a) NOx 출력 (b) Errors

그림 7. 학습 데이터에 대한 EFPNN의 출력과 오차

Fig. 7. Output and errors of EFPNN for learning dataset



(a) NOx 출력 (b) Errors

그림 8. 테스트 데이터에 대한 EFPNN의 출력과 오차

Fig. 8. Output and errors of EFPNN for testing dataset

표 7은 가스터빈 발전소의 NOx 배출공정에 적용된 기존의 모델들과 성능지수를 비교하여 보여준다. Ahn의 모델은 전체 데이터를 이용하여 PI만을 고려한 값이다. FNN은 GAs와 개선된 complex 알고리즘의 결합에 의한 하이브리드 알고리즘을 이용하여, 멤버쉽함수의 파라미터를 동조한 구조이며, Multi FNN은 클러스터를 이용한 구조로 4개의 FNN으로 구성된다. 제안된 EFPNN 구조는 전반부와 후반부와의 상호결합을 통해 각 구조의 단점을 보완하고 성능을 개선함으로써 다른 모델보다 좋은 출력특성을 가지게 된다. 또한 GAs에 의한 구조 최적화는 좀 더 효율적이고, 객관적인 그리고 대상 공정에 대해 적응할 수 있는 유연한 구조를 설계할 수 있도록 해주며, 우수한 근사화 및 예측성능을 가지는 모델을 제공한다.

표 7. 기존 모델방법과의 동정오차 비교

Table 7. Comparison of performance with other modeling methods

Model		PI	E_PI	No. of rules
Regression model[9]		17.68	19.23	
Ahn's[18]	FNN	5.835		
	AIM	8.420		
FNN[9]	Simplified	6.269	8.778	30(6+6+6+6+6)
	Linear	3.725	5.291	30(6+6+6+6+6)
Multi FNN[15]	Simplified	2.806	5.164	30(6+6+6+6+6)
	Linear	0.720	2.025	30(6+6+6+6+6)
The proposed EFPNN	Simplified	0.065	0.189	32 rules/2^{th} layer
		0.041	0.111	32 rules/5^{th} layer
	Linear	0.049	0.646	32 rules/2^{th} layer
		0.019	0.286	32 rules/5^{th} layer

5. 결 론

본 논문은 CI 기술의 한 영역으로써 효과적인 비선형 문제의 분석과 해결을 위한 고급 CI 기술인 진화론적 최적 퍼지다항식 뉴럴네트워크(EFPNN)의 구조 개발과 광범위한 설계방법론을 토의하였다. 제안된 EFPNN은 유전자 알고리즘(GAs)을 기반으로, 퍼지관계 기반 퍼지뉴럴네트워크(FNN)와 다항식뉴럴네트워크(gPNN)와의 결합에 의해 설계되는 합성 구조이다. EFPNN은 GAs를 이용하여 후반부 구조인 PNN의 구조를 최적화하고 뉴런을 생성함으로써 유연성과 정확성을 가지도록 설계된 구조이다. 기존의 FPNN은 설계자가 네트워크를 구성하는 PN의 입력변수 수, 다항식 차수 등을 경험에 의해 결정해야만 하였다. 따라서, 객관적이고 주어진 문제의 특성에 맞는 모델을 설계하지 못하였고, 최적의 모델 구축을 위해 많은 시행착오를 필요로 하였다. 그러나 제안된 EFPNN은 GAs를 이용하여 PN의 입력변수 수, 입력변수, 다항식 차수를 결정함으로써 유연성과 정확성을 가지며, 객관적인 적응 모델을 설계할 수 있다. 따라서 제안된 EFPNN의 구조는 시스템 환경의 변화에 적응하고 진화하며, 근사화와 예측(일반화) 능력이 우수한 특징을 가진다.

참 고 문 헌

[1] G. Kang and M. Sugeno, "Fuzzy Modeling", *Transactions of the Society of Instrument and Control Engineers*, Vol. 23, No. 6, pp. 106 108, 1987.

[2] S. K. Oh and W. Pedrycz, "Fuzzy Identification by Means of Auto Tuning Algorithm and Its Application to Nonlinear Systems", *Fuzzy Sets and Systems*, Vol. 115, No. 2, pp. 205 230, 2000.

[3] K. S. Narendra and K. Parthasarathy, "Gradient Methods for the Optimization of Dynamical Systems Containing Neural Networks", *IEEE Transactions on Neural Networks*, Vol. 2, pp. 252 262, 1991.

[4] W. Pedrycz and J. F. Peters, *Computational Intelligence and Software Engineering*, World Scientific, Singapore, 1998.

[5] B. J. Park, W. Pedrycz and S. K. Oh, "Fuzzy Polynomial Neural Networks: Hybrid Architectures of Fuzzy Modeling", *IEEE Transaction on Fuzzy Systems*, Vol. 10, Issue 5, pp. 607-621, 2002.

[6] S. K. Oh, W. Pedrycz and B. J. Park, "Self organizing Neurofuzzy Networks Based on Evolutionary Fuzzy Granulation", *IEEE Transaction on Systems, Man and Cybernetics part A*, Vol. 33, No. 2, pp. 271-277, 2003.

[7] S. K. Oh and W. Pedrycz, "The Design of Self-organizing Polynomial Neural Networks", *Information Sciences*, Vol. 141, Issue 3 4, pp. 237 258, 2002.

[8] S. K. Oh, W. Pedrycz and B. J. Park, "Polynomial Neural Networks Architecture: Analysis and

Design", *Computers and Electrical Engineering*, Vol. 29, Issue 6, pp. 653-725, 2003.
[9] S. K. Oh, W. Pedrycz and H. S. Park, "Hybrid Identification in Fuzzy-Neural Networks", *Fuzzy Sets and Systems*, Vol. 138, pp. 399-426, 2003.
[10] S. I. Horikawa, T. Furuhashi and Y. Uchigawa, "On Fuzzy Modeling Using Fuzzy Neural Networks with the Back Propagation Algorithm", *IEEE Transactions on Neural Networks*, Vol. 3, No. 5, pp. 801-806, 1992.
[11] David E. Goldberg, *Genetic Algorithms in search, Optimization&Machine Learning*, Addison-wesley, 1989.
[12] Z.Michalewicz, *Genetic Algorithms+Data Structure =Evolution Programs*, Springer-Verlag, 1992.
[13] A. G. Ivakhnenko, "The Group Method of Data Handling; a Rival of Method of Stochastic Approximation", *Soviet Automatic Control*, Vol. 1, No. 3, pp. 43-55, 1968.
[14] T. Kondo, "Revised GMDH algorithm estimating degree of the complete polynomial", *Transactions of the Society of Instrument and Control Engineers*, Vol. 22, No. 9, pp. 928-934, 1986.
[15] H. S. Park and S. K. Oh, "Multi-FNN Identification Based on HCM Clustering and Evolutionary Fuzzy Granulation", *International Journal of Control, Automation and Systems*, Vol. 1, No. 2, pp. 194-202, 2003.
[16] 오성권, 프로그래밍에 의한 컴퓨터지능(퍼지, 신경회로망 및 진화알고리즘을 중심으로), 내하출판사, 2002.
[17] 管野道夫(譯:박민용,최항식), 퍼지제어 시스템, pp. 143-158, 대영사, 1990.
[18] 안태천, 오성권, "발전소의 대기오염물질 배출패턴 모델정립", 기초전력공학 공동연구소, 1997.

저 자 소 개

박병준(Byoung-Jun Park)
1998년 : 원광대 제어계측공학 졸업.
2000년 : 동 대학원 졸업(공석).
2003년 : 동 대학원 졸업(공박).
2004년~2005년 : 원광대 전기전자및정보공학부 강의교수
2005년~현재 : Postdoctoral fellow in Dept. of Electrical and Computer Eng., University of Alberta.
관심분야 : CI, 뉴로퍼지컴퓨팅, 퍼지이론, 지능형 모델링 및 제어 등.

Phone : 063-850-6342
E-mail : lcap@wonkwang.ac.kr

이동윤(李東潤)
1987년 : 원광대 전기공학과 졸업.
1990년 : 연세대 대학원 전기공학과 졸(공석).
2001년 : 동 대학원 졸업(공박).
2002년~현재 : 중부대학교 정보통신S/W 공학과 조교수
관심분야 : 통신시스템, 정보통신, 지능시스템

Phone : 041-750-6209
E-mail : dylee@joongbu.ac.kr

오성권(Sung-Kwun Oh)
1983년 : 연세 대학원 졸업(공석).
1993년 : 동 대학원 졸업(공박)
1993년~2005 : 원광대 전기전자및정보공학부 교수
2005년~현재 : 수원대 전기공학과 교수
2002년~현재 : 대한전기학회 및 제어자동화시스템공학회 편집 위원
관심분야 : 뉴로퍼지컴퓨팅, 퍼지이론, 지능형 모델링 및 제어 등.

Phone : 063-850-6342
E-mail : ohsk@suwon.ac.kr