

퍼지-뉴럴 네트워크의 새로운 설계와 비선형 공정으로의 응용

A New Approach to Fuzzy-Neural Networks and Its Application to Nonlinear Processes

박호성*, 오성권**

Abstract

In this paper, we suggest a new approach method to Fuzzy-Neural Networks(FNN) model for complex and nonlinear process systems. The proposed FNNs architecture uses simplified and linear inference as fuzzy inference method and error back propagation algorithm as learning rules. And we use an Genetic Algorithms(GAs) for optimization of model. That is, the parameters such as parameters of membership functions, learning rates and momentum coefficients are adjusted using genetic algorithms. And an aggregate objective function(performance index) with weighted value is proposed to achieve a sound balance between approximation and generalization abilities of the model. According to selection and adjustment of a weighting factor of an aggregate objective function, we show that it is available and effective to design an optimal FNN model structure with a mutual balance and dependency between approximation and generalization abilities. Here, approximation and generalization abilities mean training and validation of model, respectively. In addition to training and validation, we use testing of model. To evaluate the performance of the proposed model, we use the time series data for gas furnace and the NOx emission process data of gas turbine power plant.

1. 서론

우리가 시스템을 설계할 때 항상 문제가 되는 것은 복잡성과 불확실성이며 이에 대처하는 방법에 따라 설계한 대상의 기능에 대한 성능이 가려진다. 지난 몇 년간 전통적인 선형 방정식 기반 근사화

* 원광대학교 제어계측공학과 박사과정

** 원광대학교 전기전자 및 정보공학부 부교수

이론은 현재 잘 개발되었으며 실제 문제에서 있어서도 성공적으로 실행되고 있는 반면에 비선형 함수 근사화와 같은 문제에서는 신경망과 퍼지 기술과 같은 방법을 적용하는 데 상당한 관심이 집중되고 있다[1]. 이러한 비선형 공정에 대한 복잡성과 불확실성을 제어하기 위해 퍼지 이론과 신경망과 같은 지능형 모델에 대해 많은 연구가 이루어지고 있다.

퍼지이론은 1965년 Zadeh[2]에 의해 창안된 “퍼지 집합” 이론을 이용하여 비선형적이고 복잡한 실 시스템의 특성을 해석하는데 적용함으로써 시스템의 성능 및 기능면에 있어서 기존의 수학적 모델보다 좋은 결과를 가져왔다. 인간의 두뇌가 이루고 있는 신경의 구조를 응용한 신경망(Neural networks)은 학습을 통하여 모델이 지능적으로 행동할 수 있도록 하였고 계산처리에 있어서 병렬 분산처리를 함으로써 계산처리 능력을 증가시키는 장점을 가졌다.

이에 본 논문에서는 지능형 모델의 하나로 다른 퍼지-뉴럴 네트워크모델에 비해서 학습속도가 빠르고, 수렴특성이 매우 우수하며 후반부 추론은 간략 추론을 사용한 퍼지-뉴럴 네트워크(FNN) 모델[3,4]을 기반으로 하여 후반부 추론 방법을 선형 추론 방법으로 확장한 퍼지-뉴럴 네트워크 모델을 제안한다. 이는 보수적(complementary) 멤버쉽함수를 가지는 퍼지규칙과 오류 역전파 알고리즘을 사용한 신경회로망의 학습을 결합한 형태를 사용하고, 각 입력변수에 의한 퍼지 입력공간분할에 의해 생성되는 네트워크 구조이다.

제안된 퍼지-뉴럴 네트워크모델 구조의 최적화를 위해 유전자 알고리즘[5,6]을 이용하여 모델의 멤버쉽 함수 정점과 학습을 그리고 모멘텀 계수를 최적

으로 동조한다. 더 나아가 학습 데이터(Training data) 및 확인 데이터(Validation data)를 고려한 퍼지-뉴럴네트워크 모델 성능 즉 근사화 및 일반화 능력 모두를 고려한 성능결과를 위해 하중 계수를 가진 합성 목적함수[7]를 사용한다.

모델의 성능지수는 데이터의 개수, 비선형성의 정도(입출력 데이터 분포)등에 의해 상호 의존성을 가진 근사화 및 예측의 정확도에 따라 차이가 나므로 하중계수의 선택과 조절을 통하여 공정시스템의 최적모델을 구축할 수 있다. 또한 모델의 최종적인 평가를 위해서 테스트 데이터(Testing data)를 사용하여 하중계수를 가진 합성 목적함수에 의해서 구축된 모델을 평가한다. 제안된 모델은 가스로 공정[8]과 가스터빈 발전소의 NOx 방출 공정 데이터[9]를 사용하여 성능을 평가한다.

2. 본 론

2.1 퍼지-뉴럴 네트워크의 구조

퍼지-뉴럴네트워크는 퍼지와 신경회로망의 두 구조가 유사성과 상호 보완적인 특성으로부터 결합이 시도되었다.

이 결합에 의한 퍼지-뉴럴네트워크 구조는 퍼지 이론의 언어적 논리를 규칙의 형태로 표현할 수 있는 능력과 신경회로망의 학습기능과의 상호 높은 융통성을 가지게 되는데, 본 논문에서 제안한 퍼지-뉴럴 네트워크의 모델은 그림 1에서 보이는 바와 같이 다른 모델에 비해서 학습속도가 빠르고, 수렴 특성이 매우 우수하며 후반부 추론을 간략 추론 방법을 이용한 FNN[3,4]구조를 후반부 추론방법을 선형 추론까지 확장한 FNN을 사용한다.

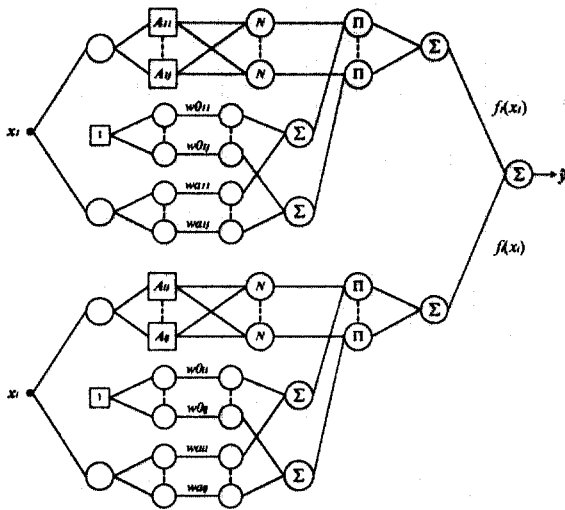


그림 1. 퍼지-뉴럴 네트워크의 구조

Fig 1. Structure of fuzzy-neural networks

본 논문에 사용한 FNN 모델은 그림 1과 같이 퍼지추론과 뉴럴네트워크의 오류 역전파 알고리즘을 합성한 형태이다. 일반적인 신경회로망과는 달리 제안된 퍼지-뉴럴 네트워크에서는 시그모이드 함수는 포함되지 않는다. 각 퍼지-뉴런에서 입력 신호 x_i 는 각 멤버쉽 함수를 거쳐 적합도가 구해지고, 적합도는 연결 가중치와 곱해지며 이들이 대수적으로 합해져 i 번째 퍼지-뉴런의 출력이 된다. 각 퍼지-뉴런에서 출력된 값들은 식 (1)과 같이 모두 대수적으로 합해져 \hat{y} 를 최종적으로 출력하게 된다.

$$\hat{y} = f_1(x_1) + f_2(x_2) + \dots + f_m(x_m) = \sum_{i=1}^m f_i(x_i) \quad (1)$$

여기서, m 은 퍼지-뉴런의 수이다.

각각의 퍼지-뉴런의 내부구조는 퍼지규칙을 이루고 있으며 그 규칙은 다음과 같이 간략 추론 방법일 경우와 선형 추론 방법일 경우로 나뉘어진다.

구조 I : 간략 추론인 경우

간략추론 경우엔 다음과 같이 퍼지규칙을 이룬다.

$$\begin{aligned} R^1 &: \text{IF } x_i \text{ is } A_{i1} \text{ Then } y_{i1} = w_{01} \\ &\vdots \\ R^j &: \text{IF } x_i \text{ is } A_{ij} \text{ Then } y_{ij} = w_{0j} \\ &\vdots \\ R^n &: \text{IF } x_i \text{ is } A_{in} \text{ Then } y_{in} = w_{0n} \end{aligned} \quad (2)$$

식 (2)에서 R^j 는 j 번째 퍼지규칙을 나타내며, A_{ij} 는 전반부의 퍼지변수이고 그림 2와 같이 멤버쉽 함수 μ_{ij} 로 표현된다. 후반부의 w_{0j} 는 상수이며, 그림 1에서 나타내듯이 연결 가중치를 나타낸다. 퍼지추론의 전반부 멤버쉽 함수는 그림 2에서 볼 수 있듯이 이웃하는 멤버쉽 함수와 보수적(complementary)인 관계를 가지는 삼각형 형태를 사용한다.

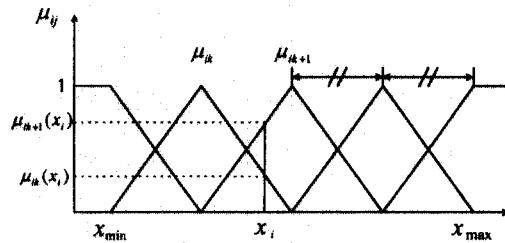


그림 2. 멤버쉽 함수 정의

Fig. 2. Definition of M.F.

x_i 에서 $f_i(x_i)$ 로의 사상(mapping)은 퍼지추론과 비퍼지화에 의해 이루어진다. 식 (2)의 최종 추론 결과는 식 (3)과 같다. 이 식은 무게 중심법(center of gravity)에 의한 비퍼지화(defuzzification) 과정까지 포함한다.

$$f_i(x_i) = \frac{\sum_{j=1}^n \mu_{ij}(x_i) \cdot w_{0j}}{\sum_{j=1}^n \mu_{ij}(x_i)} \quad (3)$$

입력변수 x_i 는 그림 2와 같이 동시에 두개의 멤버쉽 함수 $k, k+1$ 에만 영향을 받기 때문에 식 (3)은 다음과 같이 다시 쓸 수 있다.

$$f_i(x_i) = \frac{\mu_{ik}(x_i) \cdot u0_{ik} + \mu_{ik+1}(x_i) \cdot u0_{ik+1}}{\mu_{ik}(x_i) + \mu_{ik+1}(x_i)} \quad (4)$$

보수적 관계를 가지는 2개의 이웃하는 멤버쉽 함수의 입력변수에 대한 적합도 합은 언제나 1이 되므로, $\mu_{ik}(x_i) + \mu_{ik+1}(x_i) = 1$ 이 된다. 따라서, 다시 식 (5)와 같이 쓸 수 있다.

$$f_i(x_i) = \mu_{ik}(x_i) \cdot u0_{ik} + \mu_{ik+1}(x_i) \cdot u0_{ik+1} \quad (5)$$

이와 같은 특징 때문에 추론속도가 증가하는 효과를 얻을 수 있다.

구조 II : 선형 추론인 경우

선형추론 경우엔 다음과 같이 퍼지규칙을 이룬다.

$$\begin{aligned} R^1 : IF \ x_i \text{ is } A_{i1} \ \text{Then} \ y_{i1} &= u0_{i1} + x_i wa_{i1} \\ &\vdots \\ R^j : IF \ x_i \text{ is } A_{ij} \ \text{Then} \ y_{ij} &= u0_{ij} + x_i wa_{ij} \quad (6) \\ &\vdots \\ R^n : IF \ x_i \text{ is } A_{in} \ \text{Then} \ y_{in} &= u0_{in} + x_i wa_{in} \end{aligned}$$

식 (6)의 최종 추론결과는 식 (7)과 같다.

$$f_i(x_i) = \frac{\sum_{j=1}^n \mu_{ij}(x_i) \cdot (u0_{ij} + x_i wa_{ij})}{\sum_{j=1}^n \mu_{ij}(x_i)} \quad (7)$$

간략 추론일 경우와 마찬가지로 식(7)은 식 (8)과 같이 다시 쓸 수 있다.

$$f_i(x_i) = \frac{\mu_{ik}(x_i) \cdot (u0_{ik} + x_i wa_{ik}) + \mu_{ik+1}(x_i) \cdot (u0_{ik+1} + x_i wa_{ik+1})}{\mu_{ik}(x_i) + \mu_{ik+1}(x_i)} \quad (8)$$

2.2 퍼지-뉴럴 네트워크의 학습

퍼지-뉴럴네트워크의 학습은 그림 1의 연결 가중치를 모델에 가장 적절한 값으로 조정해 나가면서 이루어진다. 학습방법은 신경회로망의 학습에 가장 일반적으로 사용되는 오류 역전파 알고리즘이 사용되는데, 이는 최종출력과 원 출력 데이터간의 차, 즉 오차를 바탕으로 이루어진다. 본 논문에서는 식 (9)와 같은 형태의 출력오차를 사용한다.

$$E_p = (y_p - \hat{y}_p)^2 \quad (9)$$

여기서 E_p 는 각 데이터에 대한 오차를 의미하며, y_p 는 주어진 출력 데이터, \hat{y}_p 는 모델의 출력데이터를 나타낸다. m 개의 입출력 데이터 쌍이 주어진 경우, 퍼지-뉴럴 네트워크의 최종 출력 오차는 식 (10)과 같이 된다.

$$E = \frac{1}{m} \sum_{p=1}^m E_p \quad (10)$$

연결 가중치의 조절은 식 (11)과 같이 기존값과 오차로부터 구해진 조절량 만큼의 변화를 더해주면 된다.

$$w(new) = w(old) + \Delta w \quad (11)$$

오차 E_p 를 최소화하기 위한 연결 가중치의 변화량 Δw 는 식 (12)에서 보이는 것처럼 경사강하법 (gradient descent method)을 이용하여 구할 수 있다.

$$\Delta w_{ij} = \eta \left(- \frac{\partial E_p}{\partial w_{ij}} \right) \quad (12)$$

여기서, η 는 학습율이다.

I) 상수항인 경우

식 (11)과 식(12)를 다시 쓰면

$$u0(new) = u0(old) + \Delta u0 \quad (13)$$

$$\Delta u0_{ij} = \eta \left(- \frac{\partial E_p}{\partial u0_{ij}} \right) \quad (14)$$

식 (14)에서 우변의 편미분은 사슬법칙에 의해 식 (15)와 같이 세 부분으로 전개될 수 있다.

$$- \frac{\partial E_p}{\partial u0_{ij}} = - \frac{\partial E_p}{\partial \hat{y}_p} \cdot \frac{\partial \hat{y}_p}{\partial f_i(x_i)} \cdot \frac{\partial f_i(x_i)}{\partial u0_{ij}} \quad (15)$$

식 (15)에서 우측 식의 각 항은 다음 식 (16)~(18)과 같이 유도된다.

$$- \frac{\partial E_p}{\partial \hat{y}_p} = 2(y_p - \hat{y}_p) \quad (16)$$

$$\frac{\partial \hat{y}_p}{\partial f_i(x_i)} = 1 \quad (17)$$

$$\frac{\partial f_i(x_i)}{\partial u_{ij}} = \mu_{ij}(x_i) \quad (18)$$

그러므로 변화량, Δu_{ij} 는 식 (16)~(18)으로부터 식 (19)와 같이 된다.

$$\Delta u_{ij} = 2 \cdot \eta_0 \cdot (y_p - \hat{y}_p) \cdot \mu_{ij}(x_i) \quad (19)$$

여기에 모멘텀을 더해 학습시간을 좀 더 단축시키고, 또한 학습성능을 향상시킨다. 모멘텀은 관성이란 뜻으로, 연결 가중치의 변화량이 클 경우에는 좀 더 변화량을 키워주고, 변화량이 작아지면 또 여기에 맞춰서 변화량을 좀 줄여주는 역할을 한다. 모멘텀은 식 (20)과 같이 현재의 연결 가중치와 그 이전 연결 가중치의 차이에 의해 계산된다.

$$m(t) = w(t) - w(t-1) \quad (20)$$

이렇게 계산된 모멘텀은 모멘텀 계수라 불리는 α 와 곱해져 연결 가중치 조절식에 추가된다. 모멘텀까지 포함하는 연결 가중치의 최종 조절식은 식 (21)과 같다.

$$\Delta u_{ij} = 2 \cdot \eta_0 \cdot (y_p - \hat{y}_p) \cdot \mu_{ij}(x_i) + \alpha_0 (u_{ij}(t) - u_{ij}(t-1)) \quad (21)$$

II) 변수형인 경우

식 (11)과 식(12)를 다시 쓰면

$$wa(\text{new}) = wa(\text{old}) + \Delta wa \quad (22)$$

$$\Delta wa_{ij} = \eta_a \left(-\frac{\partial E_p}{\partial wa_{ij}} \right) \quad (23)$$

식 (23)는 식 (24)와 같이 세 부분으로 전개될 수 있다.

$$-\frac{\partial E_p}{\partial wa_{ij}} = -\frac{\partial E_p}{\partial \hat{y}_p} \cdot \frac{\partial \hat{y}_p}{\partial f_i(x_i)} \cdot \frac{\partial f_i(x_i)}{\partial wa_{ij}} \cdot x_i \quad (24)$$

식 (24)에서 우측 식의 각 항은 상수항의 경우인 식 (16)-(18)와 마찬가지로여서 최종적인 변화량은 식

(25)와 같이 된다.

$$\Delta wa_{ij} = 2 \cdot \eta_a \cdot (y_p - \hat{y}_p) \cdot \mu_{ij}(x_i) \cdot x_i \quad (25)$$

모멘텀까지 포함하는 연결 가중치의 최종 조절식은 식 (26)과 같다.

$$\Delta wa_{ij} = 2 \cdot \eta_a \cdot (y_p - \hat{y}_p) \cdot \mu_{ij}(x_i) \cdot x_i + \alpha_a (wa_{ij}(t) - wa_{ij}(t-1)) \quad (26)$$

3. 유전자 알고리즘과 합성 목적함수를 이용한 모델의 최적화

3.1 유전자 알고리즘

Holland[10]는 파라미터 공간의 동정에 있어서 높은 성능 영역을 위해 지역 탐색 루틴에 의해 수행되는 유전자 알고리즘을 제안했다. 후에 지역 탐색 능력은 형식이 없는 돌연변이에 의해 개선되었다 [6]. 최적화 문제에서 탁월한 성능을 발휘하는 유전자 알고리즘[5,6]은 생물학적 진화, 자연선택, 유전자 재조합, 군집 도는 개체에서 "적자생존"의 생물학 원리에 바탕을 둔 확률적 탐색 방법이다. 또한 기존의 방법들과는 달리 선형, 연속, 미분가능 등의 제한이 없기 때문에 다양한 분야에 별다른 제한 없이 적용할 수 있다는 장점을 가진다. 최근의 많은 탐색방법들은 한 번에 최적값을 찾아내지 않고, 반복을 통해서 최적지점에 접근해 가는 방법을 사용하고 있다. 유전자 알고리즘도 마찬가지로 적정 수준에 도달할 때까지 탐색과정을 반복한다. 제안한 모델의 최적화를 위하여 사용된 유전자 알고리즘은 반복을 통해서 최적지점에 접근해 가는 방법을 사용하고 탐색지점으로 실수가 주어지면 이를 2진수(binary)로 변환하게 된다. 변환된 각 비트는 염색체처럼 취급되고, 염색체가 모여 유전자를 이루며

개체(individual)라 불린다. 즉, 각 염색체는 그림 3과 같이 모두 하나의 개체 안에 연결해 사용한다. 염색체(변수)는 10 bit로 정의하고, 개체가 탐색할 때 사용되는 하나의 지점을 의미한다. 동시에 여러 점을 탐색하기 위해 개체를 동시에 여러 개 사용되는데 이를 군집이라 한다.

본 논문에서는 퍼지-뉴럴네트워크의 모델에 대한 멤버쉽 함수의 정점과 학습율, 모멘텀 계수를 최적으로 설정해 주기 위하여 유전자 알고리즘을 이용하여 모델의 최적화 과정을 수행하게 된다.

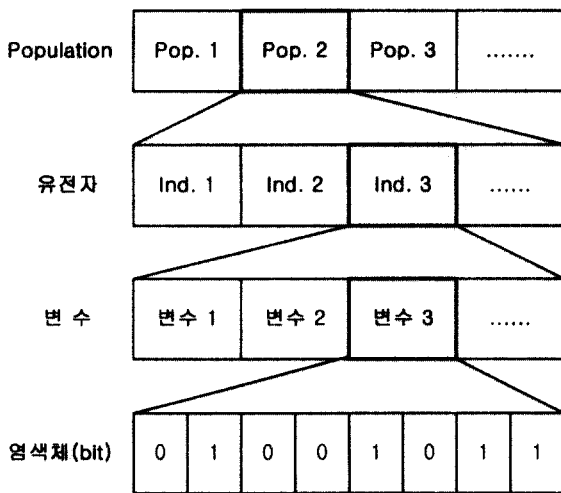


그림 3. 유전자 알고리즘 데이터 구조의 계층 표현
Fig. 3. Class expression of genetic algorithms data structure

표 1은 유전자 알고리즘을 시뮬레이션하기 위해 설정된 각각의 개체에 존재하는 염색체의 수와 군집의 크기에 대한 파라미터의 정의으로써, 한 개체에 여러 변수의 정보를 담은 직렬 방식을 사용한다. 선택(selection)연산으로 roulette-wheel방식, 돌연변이 연산은 선택된 비트를 반전시키는 invert방식, 교차(crossover)연산은 one-crossover방식을 사용하였다.

표 1. 유전자 알고리즘의 파라미터

Table 1. Parameter of genetic algorithm

| | |
|--------------------|------|
| Population size | 60 |
| Maximum generation | 100 |
| Crossover rate | 0.6 |
| Mutation rate | 0.35 |

3.2 하중 계수를 가진 합성 목적함수

입출력 데이터로 시스템을 모델링 할 때 가장 문제가 되는 것은 구축된 모델이 학습 데이터만을 사용한 모델로써 시스템을 구축하는 경우이다. 이는 대상 공정의 모델을 주어진 데이터에 의해서만 성능향상을 꾀하였기 때문에 실 공정에 적용하면 많은 오차를 발생시키고 정확한 예측능력을 가지지 못한다. 이러한 현상은 비선형 공정인 경우 더욱 문제가 된다. 이 문제를 해결하기 위하여 여러 방법이 제안되고 있는데 그 중에서 학습 데이터를 사용하여 모델을 생성한 후 그 모델에 테스트 데이터를 적용시켜 가장 우수한 성능지수를 가지는 모델을 최적모델로 선정하였다. 그러나 이것 또한 공정모델에 대한 성능 평가를 정확히 반영하지 못한다. 따라서 본 논문에서는 주어진 학습 데이터를 이용하여 퍼지-뉴럴네트워크 모델을 구축하고 확인 데이터를 사용하여 모델을 평가하기 위하여 근사화 및 일반화의 상호 균형과 의존 능력을 가진 합성 목적 함수(성능지수)를 이용하여 상호 연계된 최적 모델 구조를 생성한다[7]. 이를 위해 학습 및 확인 데이터의 성능 결과에 대한 하중계수를 가진 목적함수인 합성 성능지수로써 식 (27)과 같이 정의한다.

$$f(PI, V_PI) = \theta \times PI + (1 - \theta) \times V_PI \quad (27)$$

θ 는 PI 와 V_PI 에 대한 하중계수를 나타내고, PI 는 학습 데이터에 대한 성능지수, V_PI 는 확인 데

이터에 대한 성능지수를 나타낸다. 그리고 목적함수는 모델의 성능향상이 최적이 되도록 하기 위해 각각의 퍼지-뉴럴 네트워크의 멤버십 함수의 정점들과 학습을, 모멘텀 계수들은 유전자 알고리즘에 의해 조절된다. 식 (27)과 같이 정의한 하중계수를 가진 목적함수는 하중계수의 설정에 따라 다음과 같은 특징을 가진다.

첫째, $\theta=1$ 이면 목적함수는 $f(PI, V_PI) = PI$ 와 같으므로 학습 데이터를 기초로 최적화된다. 이것은 단지 학습 데이터에 의한 퍼지-뉴럴 네트워크 모델의 성능지수 PI 를 목적함수로 놓기 때문에 모델의 근사화 능력은 뛰어나지만, 일반화 능력(즉, 예측능력)이 떨어진다.

둘째, $\theta=0$ 이면 목적함수는 $f(PI, V_PI) = V_PI$ 와 같으므로 학습 데이터를 이용하여 모델링한 후 확인 데이터를 기초로 최적화된다. 이것은 단지 확인 데이터에 의한 퍼지-뉴럴 네트워크 모델의 성능지수 V_PI 를 목적함수로 놓기 때문에 첫 번째 경우보다는 모델의 근사화 능력은 떨어질지 모르나, 일반화 능력은 더욱 향상된다.

셋째, $\theta=0.5$ 이면 목적함수는 $f(PI, V_PI) = 0.5PI + 0.5V_PI$ 와 같으므로 학습과 확인 능력 모두 같은 비중을 가지고 평가한다. 이것은 첫 번째 특징과 비교해볼 때 근사화 능력은 떨어질지 모르나 일반화 능력은 향상되며, 두 번째 특징과 비교해보면 근사화 능력은 우수하나 일반화 능력이 떨어진다는 특징을 가지고 있다.

넷째, $\alpha \in [0, 1]$ 에 대해서 $\theta = \alpha$ 이면 학습과 확인 능력 모두를 포함하고, α 의 선택은 모델의 근사화와 일반화 사이에서 상호 균형에 의한 최적화를 시도한다. 이 의미는 학습 데이터에 의한 퍼지-뉴

럴 네트워크모델의 성능지수 PI 를 구하고, 학습 데이터에 의해 생성된 퍼지-뉴럴 네트워크에 확인 데이터를 가하여 평가용 성능지수 V_PI 를 구하는 것이다. 구해진 PI 와 V_PI 각각에 하중계수 θ 와 $1-\theta$ 을 곱한 다음, 합에 의한 목적함수 $f(\cdot)$ 을 최적화시키는 방향으로 모델의 파라미터를 동조함으로써 최적 퍼지-뉴럴 네트워크를 구축한다. 이것은 모델 자체의 근사화 능력과 일반화 능력을 상호 고려한 목적함수의 형태로 적절한 하중계수 선정에 따라 최적 퍼지-뉴럴네트워크 모델을 구축할 수 있다.

4. 시뮬레이션 및 결과고찰

4.1 가스로 공정

제안된 퍼지-뉴럴 네트워크의 성능을 평가하기 위해서 많은 연구자들이 제안하는 모델링 기법의 성능을 평가하는데 널리 사용되는 가스로 시계열 데이터를 이용하였다. Box와 Jenkins[8]에 의해 사용된 가스로 시계열 데이터는 입출력 데이터인 가스 흐름율과 연소된 이산화탄소 농도의 가스로 공정으로부터 추출된다. 메탄가스 흐름율($X(t)$)은 실험에서 -2.5에서 2.5로 변하는 것을 사용하나 실제 공정에서의 메탄가스 흐름율($X_m(t)$)은 식 (28)에 의해 0.5에서 0.7로 변한다.

$$X_m(t) = 0.6 - 0.048 \times X(t) \quad (28)$$

입력($u(t)$)은 가스 흐름율이고 출력($y(t)$)은 이산화탄소 농도인 1입력 1출력의 가스로 시계열 데이터 296쌍을 시뮬레이션 하기 위해 균일한 입출력 분포 구조를 가지는 입력 $u(t-3)$, $y(t-1)$ 과 출력 $y(t)$ 의 2입력 1출력으로 설정하여 사용한다. 또한 전체 데이

터 중 처음부터 125번째의 입력 데이터의 집합을 학습 데이터로, 126번째부터 250번째 데이터를 확인 데이터로 그리고 251번째 데이터부터 마지막 데이터를 테스트 데이터로 나누어 퍼지-뉴럴 네트워크를 모델링하고 유전자 알고리즘과 하중계수를 가진 합성 목적함수를 사용하여 모델의 최적화한다.

표 2에서는 θ 값 설정에 따라 FNN 모델의 멤버쉽 함수의 파라미터들과 학습율, 모멘텀 계수를 최적화 알고리즘에 의해 자동 동조함으로써 최적 FNN 모델의 성능지수를 나타낸 것이다.

표 2. 멤버쉽 함수의 개수에 따른 성능지수

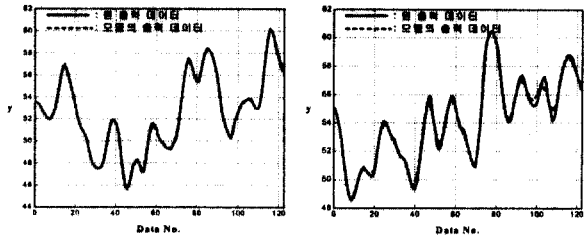
Table 2. Performance index according to the No. of M.F.

| θ | 멤버쉽 함수의 개수 | | | | | |
|----------|------------|-------|-------|-------|-------|-------|
| | 2 + 2 | | | 3 + 3 | | |
| | PI | V_PI | E_PI | PI | V_PI | E_PI |
| 0.0 | 0.159 | 0.094 | 0.794 | 0.047 | 0.098 | 0.830 |
| 0.25 | 0.027 | 0.098 | 0.846 | 0.024 | 0.098 | 0.852 |
| 0.5 | 0.024 | 0.099 | 0.845 | 0.023 | 0.099 | 0.858 |
| 0.75 | 0.023 | 0.101 | 0.852 | 0.022 | 0.099 | 0.856 |
| 1.0 | 0.023 | 0.101 | 0.853 | 0.022 | 0.100 | 0.862 |

| θ | 선형추론 멤버쉽 함수의 개수 | | | | | |
|----------|-----------------|-------|-------|-------|-------|-------|
| | 2 + 2 | | | 3 + 3 | | |
| | PI | V_PI | E_PI | PI | V_PI | E_PI |
| 0.0 | 0.024 | 0.097 | 0.824 | 0.026 | 0.095 | 0.834 |
| 0.25 | 0.024 | 0.097 | 0.833 | 0.025 | 0.097 | 0.869 |
| 0.5 | 0.024 | 0.097 | 0.833 | 0.024 | 0.097 | 0.838 |
| 0.75 | 0.023 | 0.097 | 0.839 | 0.023 | 0.099 | 0.875 |
| 1.0 | 0.023 | 0.100 | 0.849 | 0.023 | 0.097 | 0.861 |

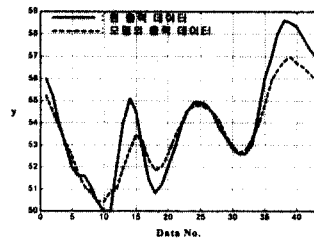
표 2에서 볼 수 있듯이 후반부가 간략 추론을 사용할 때 보다 선형 추론을 사용한 경우에서 성능이 좀 더 우수함을 알 수 있으며 또한 선형 추론을 사용할 경우 각 입력에 대한 멤버쉽 함수의 수는 2개를 사용할 경우가 3개를 사용한 경우보다 우수함을 알 수 있다.

그림 4는 선형 추론을 사용하고 각 입력의 멤버쉽 함수의 수를 2개로 설정했을 때 각각의 원 출력 데이터와 모델의 출력 데이터를 나타낸 것이다. 이 때 하중계수(θ)는 0.75일 때이다.



(a) PI

(b) V_PI



(c) E_PI

그림 4. 모델의 출력 곡선

Fig. 4. Output curves of model

그림 5는 그림 4에서 보여준 모델의 PI, V_PI, E_PI의 성능지수들의 에러 곡선을 나타낸 것이다.

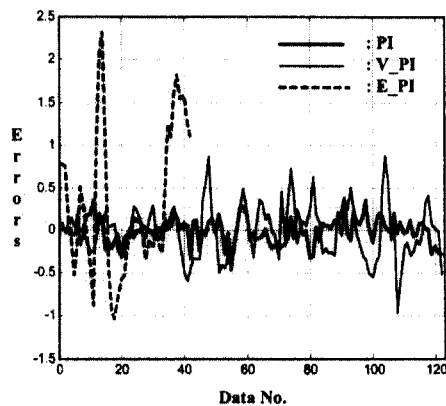


그림 5. 모델의 에러 곡선

Fig. 5. Errors curves of model

그림 6은 그림 5에서 보여준 PI, V_PI, E_PI 각각의 성능지수들이 유전자 알고리즘에 의해 최적화되는 과정을 보여준다.

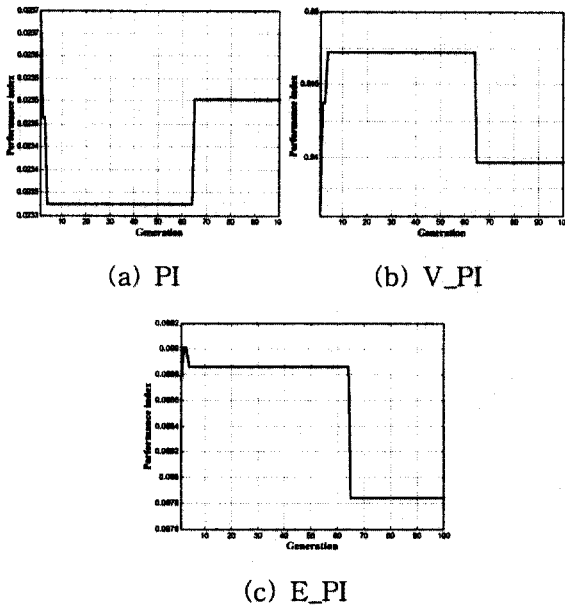


그림 6. 유전자 알고리즘에 의한 FNN 모델의 각 성능지수의 최적화 과정

Fig. 6. The optimization process of each performance index for FNN model by Genetic algorithms

4.1 가스터빈 발전소의 NOx 방출 공정 데이터

대기상 질소산화물(NOx)은 주로 연소공정을 통해 배출되며 산업공정, 발전소, 자동차 배기가스 그리고 난방에 의해서 발생하는데, 그 중 대기에서 문제가 될 만큼 존재하는 것들은 N₂O, NO 및 NO₂이고 이들 물질을 대기오염 측면에서 NOx(질소 산화물)로 일컫는다.

NOx의 저감방법으로는 발생원으로부터 NOx의 배출을 억제하는 방법과 배출가스중의 NOx를 제거하는 방법을 들 수 있는데, 최근에는 Gas Turbine

에 사용하는 입출력 변수 중에 입력으로 Tamb(The ambient temperature at site, in degrees F), COM(The compressor speed, in rpm), LPT(The low pressure turbine speed, in rpm), Pcd(The compressor discharge pressure, in psia), Texh(The turbine exhaust temperature, in degrees F)등의 주변 환경변수를 조절하며 출력으로는 Parts per million-volume dry 단위의 NOx양으로 배출 메카니즘을 모델링하고, 최적 모델을 구축하여 시뮬레이터를 구성함으로써 NOx를 효과적으로 관리하는 방법이 연구되고 있다[9].

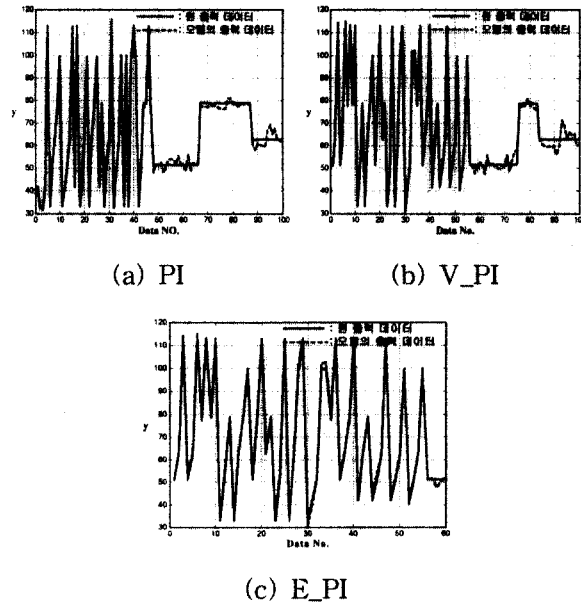
본 논문에서는 이러한 입출력 변수들을 이용하여 제안한 모델의 성능을 평가하기 위해서 각각의 변수들의 데이터에 대해 처음 100개 데이터는 학습 데이터로 사용하고 101번째부터 200번째의 데이터를 확인 데이터로 그리고 나머지 60개 데이터는 테스트 데이터로 사용하여 모델의 근사화와 일반화를 얻는다. 표 3에서는 θ 값 설정에 따라 FNN 모델의 멤버십 함수의 파라미터들과 학습율, 모멘텀 계수를 최적화 알고리즘에 의해 자동 동조함으로써 최적 FNN 모델의 성능지수를 나타낸 것이다. 이때 각 입력 변수에 대한 멤버십 함수의 수는 6이다.

표 2. 후반부 추론 방법에 따른 성능지수

Table 2. Performance index according to the consequence inference method of fuzzy rule

| θ | 간략 추론 | | | 선형 추론 | | |
|----------|-------|--------|-------|-------|-------|-------|
| | PI | V_PI | E_PI | PI | V_PI | E_PI |
| 0.0 | 6.773 | 10.827 | 6.199 | 3.371 | 6.390 | 4.852 |
| 0.25 | 6.787 | 11.132 | 6.208 | 4.343 | 6.169 | 3.766 |
| 0.5 | 7.935 | 13.079 | 5.375 | 3.810 | 5.782 | 2.457 |
| 0.75 | 6.385 | 12.055 | 7.131 | 3.660 | 6.940 | 4.196 |
| 1.0 | 6.511 | 12.334 | 7.364 | 3.537 | 5.820 | 2.874 |

그림 7은 선형 추론을 사용했을 때 각각의 원 출력데이터와 모델의 출력 데이터를 나타낸 것이다. 이때 하중계수(θ)는 0.5일 때이다.



(a) PI (b) V_PI (c) E_PI

그림 7. 모델의 출력 곡선

Fig. 7. Output curves of model

그림 8은 그림 7에서 보여준 모델의 PI, V_PI, E_PI의 성능지수들의 에러 곡선을 나타낸 것이다.

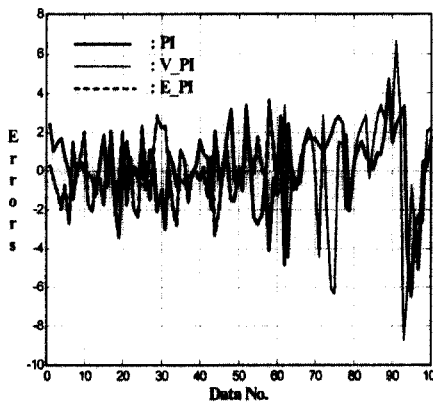
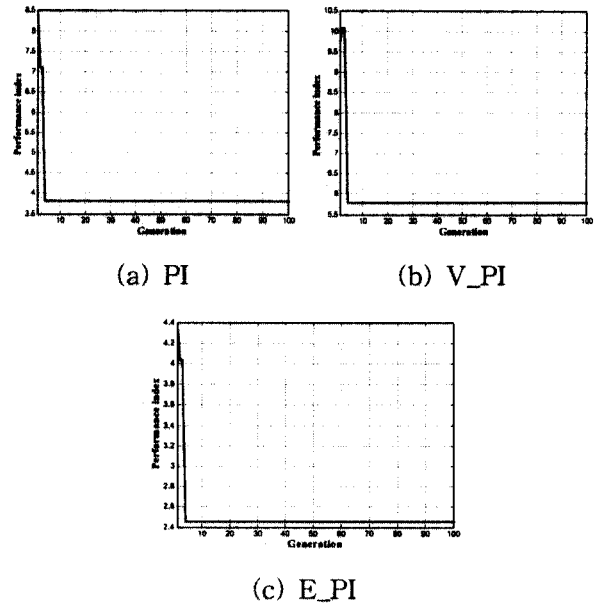


그림 8. 모델의 에러 곡선

Fig. 8. Errors curves of model

그림 9는 그림 8에서 보여준 PI, V_PI, E_PI 각각의 성능지수들이 유전자 알고리즘에 의해 최적화되는 과정을 보여준다.



(a) PI (b) V_PI (c) E_PI

그림 9. 유전자 알고리즘에 의한 FNN 모델의 각 성능지수의 최적화 과정

Fig. 9. The optimization process of each performance index for FNN model by Genetic algorithms

5. 결론

본 논문에서는 유전자 알고리즘과 하중 계수를 가진 합성 목적함수를 이용하여 최적의 퍼지-뉴럴 네트워크 모델을 구축하는데 있어서 효과적인 동정 기법을 보였다. 기존의 퍼지-뉴럴 네트워크와는 달리 후반부 추론 방법을 간략 추론뿐만 아니라 선형 추론으로 확대하여 더욱 정교하고 더 정확한 예측 능력을 가진 모델을 구축할 수 있었다. 또한 모델의

최적화를 위하여 유전자 알고리즘을 사용하여 모델의 멤버쉽 함수의 정점과 학습을, 모멘텀 계수를 동조하였으며, 하중 계수를 가진 합성 목적함수 의해 FNN 모델의 근사화 및 일반화사이에서 최적 성능 결과에 대한 해석을 시도함으로써 FNN모델의 근사화와 일반화 사이의 상호 연계를 통한 최적화 향상을 제시하였다. 더불어 합성 목적함수에 의해 구해진 최적의 모델에 기존의 방법과는 달리 테스트 데이터를 한번 더 적용함으로써 실 공정의 일반화를 더욱 강화하여 좀 더 강인한 모델을 구축할 수 있었다.

감사의 글

본 논문은 2001년 에너지 관리공단의 에너지기술 학술 진흥 사업 연구비에 의하여 지원되었음

【참 고 문 헌】

[1] J. H. Nie and T. H. Lee, Rule-based modeling: Fast construction and optimal manipulation, IEEE Trans. Syst., Man, Cybern. Part A, vol.26, pp. 728-738, Nov, 1996.

[2] L. A Zadeh, "Fuzzy sets", Inf. Control 8, 338-353, 1965.

[3] Takeshi Yamakawa, "A Neo Fuzzy Neuron and Its Applications to System Identification and Prediction of the System Behavior", Proceedings of the 2nd International Conference on Fuzzy logic & Neural Networks, pp. 477-483, 1992.

[4] Takeshi Yamakawa, "A New Effective Learning Algorithm for a Neo Fuzzy Neuron Model", 5th IFSA World Congress, pp. 1017-1020, 1993.

[5] David E. Golderg, "Genetic Algorithm in search, Optimization & Machine Learning", Addison wesley, 1989.

[6] Zbigniwe Michalewicz, "Genetic Algorithms + Data Structures = Evolution Programs", Springer-Verlag, 1992.

[7] 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 Syst., Vol. 115, No. 2, pp. 205-230, 2000.

[8] Box and Jenkins, "Time Series Analysis, Forecasting and Control", Holden Day, SanFrancisco, CA.

[9] G. Vachtsevanos, V. Ramani, T. W. Hwang, "Prediction of Gas Turbine NOx Emissions using Polynomial Neural Network", Technical Report, Georgia Institute of Technology, Atlanta, 1995.

[10] Holland, J. H. "Adaptation in Natural and Artificial Systems-An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence", The MIT Press, Cambridge, MA, 1992.