

Genetically Optimized Neurofuzzy Networks : Analysis and Design

朴炳俊* · 金賢起** · 吳聖權***

(Byoung-Jun Park · Hyun-Ki Kim · Sung-Kwun Oh)

Abstract - In this paper, new architectures and comprehensive design methodologies of Genetic Algorithms(GAs) based Genetically optimized Neurofuzzy Networks(GoNFN) are introduced, and a series of numeric experiments are carried out. The proposed GoNFN is based on the rule-based Neurofuzzy Networks(NFN) with the extended structure of the premise and the consequence parts of fuzzy rules being formed within the networks. The premise part of the fuzzy rules are designed by using space partitioning in terms of fuzzy sets defined in individual variables. In the consequence part of the fuzzy rules, three different forms of the regression polynomials such as constant, linear and quadratic are taken into consideration. The structure and parameters of the proposed GoNFN are optimized by GAs. GAs being a global optimization technique determines optimal parameters in a vast search space. But it cannot effectively avoid a large amount of time-consuming iteration because GAs finds optimal parameters by using a given space. To alleviate the problems, the dynamic search-based GAs is introduced to lead to rapidly optimal convergence over a limited region or a boundary condition. In a nutshell, the objective of this study is to develop a general design methodology of GAs-based GoNFN modeling, come up a logic-based structure of such model and propose a comprehensive evolutionary development environment in which the optimization of the model can be efficiently carried out both at the structural as well as parametric level for overall optimization by utilizing the separate or consecutive tuning technology. To evaluate the performance of the proposed GoNFN, the models are experimented with the use of several representative numerical examples.

Key Words : Genetically Optimized Neurofuzzy Networks, CI, Genetic Algorithms, Rule-based Neurofuzzy Networks

1. 서 론

인간의 일상생활 속에서는 다양한 지식이나 정보를 사용한 학습, 판단, 예측 등이 민감하게 이루어지고 있다. 많은 연구자들은 이를 하나의 함수 또는 관계로 나타내어, 인간의 판단과 유사한 결과를 도출할 수 있는 장치를 마련하고자 노력을 기울이고 있다. 최근에는 CI 기술의 바탕인 지능관련 컴퓨터 기술이 산업 전반에 다양하게 활용되면서 이의 능력에 대해 많은 관심이 모아지고 있으며, 활발한 연구를 바탕으로 여러 분야에 응용되고 있다[1,20]. CI 기술은 인간의 신경구조를 모방한 신경망[2,3]과 인간의 언어적 지식과 전문가적 경험을 사용하는 퍼지이론[4-6] 그리고 유전자 알고리즘[7,8,21]과 같은 지능시스템들이 보다 활발하게 연구되면서 합성, 융합, 결합을 통한 고급 CI 기술에 관한 연구[9-13]가 다양한 형태로 추진되고 있다.

본 논문은 고급 CI 기술의 한 영역으로써 효과적인 비선형 문제의 분석과 해결을 위한 진화론적 최적 뉴로퍼지 네트워크

(Genetically optimized Neurofuzzy Networks; GoNFN) 개발을 목표로 하며, 구조 개발을 위한 광범위한 설계방법론을 토의한다. 제안된 GoNFN은 기존의 퍼지추론과 신경망의 융합에 의해 설계된 규칙베이스 뉴로퍼지 네트워크(Rule based Neurofuzzy Networks; NFN[13-15])를 기반으로 네트워크의 결합 및 확장을 통해 상수, 선형, 그리고 이차식과 같은 회귀 다항식의 세 가지 다른 형태를 고려한다. 또한 NFN의 퍼지 규칙 후반부를 하나의 형태가 아닌 서로 다른 형태로 다양하게 표현할 수 있으며, 주어진 문제에 유연한 구조를 제공한다. 본 논문의 최종 목표를 위해 기존의 GAs를 개선한 동적 탐색기반 GAs(Dynamic search based GAs)를 제안한다. 동적 탐색기반 GAs는 기존 해를 바탕으로 탐색공간을 변화시킴으로써 주어진 문제 해결에 효율적이고, 효과적인 최적 해를 제공한다. 제안된 NFN과 동적 탐색기반 GAs를 바탕으로 구조 및 파라미터의 최적화 문제를 다루며, 최적의 GoNFN을 설계한다.

제안된 GoNFN의 성능 평가를 위해, 모델은 몇 가지 수치 예제를 이용한다. 비교에 의한 해석은 제안된 GoNFN이 규칙 수, 추정되어질 파라미터 수, 학습횟수(또는 세대) 그리고 계산상의 변수와 같은 다양한 파라미터의 관점에서 이전에 제시된 다른 지능형 모델보다 높은 정확도뿐만 아니라 좀 더 우수한 예측능력을 가지는 모델임을 보여준다.

* 正 會 員 : 圓光大學 電氣電子 · 情報工學部 · 工博

** 正 會 員 : 水原大學 電氣工學科 教授 · 工博

*** 正 會 員 : 圓光大學 電氣電子 · 情報工學部 副教授 · 工博

接受日字 : 2004年 3月 9日

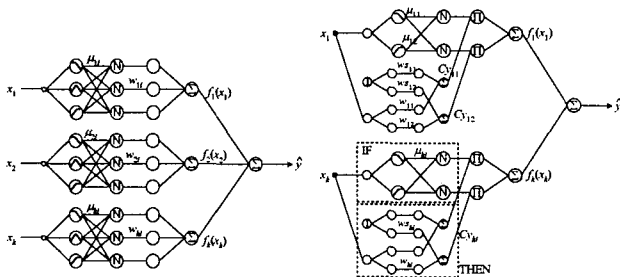
最終完了 : 2004年 5月 29日

2. 규칙베이스 뉴로퍼지 네트워크

규칙베이스 뉴로퍼지 네트워크(NFN)는 복잡성과 불확실성을 나타내는 비선형 시스템에서 다른 지능형 모델에 비해 보다 나은 특성을 가지고 있다. 이는 퍼지논리와 신경회로망이 갖는 독특한 특성을 결합한 형태로 두 이론의 시너지 효과로부터 모델의 효율과 성능을 향상시킨 모델 구조로 융합한 형태이다. 본 논문에서 다루어질 NFN은 각각의 입력변수들에 대해 정의된 퍼지집합을 기반으로 수행된다. 이것은 각 입력 변수들에 대해 독립적으로 이루어지며, 분할된 퍼지공간을 통해 퍼지규칙을 형성함으로써 NFN을 수행한다[13-15].

2.1 기존의 뉴로퍼지네트워크

퍼지집합을 기반으로 구현되는 기존의 NFN은 그림 1과 같이 퍼지-뉴런의 특성을 나타내는 비선형함수 $f(x_k)$ 로 구성된 네트워크 구조를 가지며, 이 $f(x_k)$ 은 퍼지집합에 의한 퍼지 규칙에 의해 구현된다. NFN 구조는 퍼지규칙의 추론구조에 따라 간략 퍼지추론 구조와 선형 퍼지추론 구조로 설계된다.



(a) Simplified fuzzy inference (b) Linear fuzzy inference
 그림 1 기존 NFN의 구조[13-15]

Fig. 1 Architecture of conventional NFN[13-15]

그림 1에서 ‘원’은 NFN의 뉴런 유닛을 나타내고, ‘N’은 정규화를, ‘Σ’은 대수적 합을, ‘Π’의 뉴런은 입력신호들의 곱을 생성한다. x_k 의 각 입력신호는 퍼지집합에 의해 정의된 각 멤버쉽함수를 거쳐 적합도가 구해지고, 정규화된 적합도는 간략 퍼지추론의 경우 연결 가중치와, 선형 퍼지추론의 경우 연결가중치에 의한 Cy_{ki} 와 곱해지며 이들이 Σ 뉴런에 의해 대수적으로 합해져 $f_k(x_k)$ 를 생성한다. 생성된 출력 값은 식(1)과 같이 모두 합해져 \hat{y} 를 출력하게 된다. k 는 입력변수 수이다.

$$\hat{y} = f_1(x_1) + f_2(x_2) + \dots + f_k(x_k) = \sum_k f_k(x_k) \quad (1)$$

입력 x_k 에 대한 $f_k(x_k)$ 는 퍼지규칙을 이루고 있으며, 그 규칙은 식(2)와 식(3)과 같다. R^i 는 i 번째 퍼지규칙을 나타내며, A_{ki} 는 전반부 멤버쉽함수로 보수적(complementary)인 관계를 가진 삼각형 형태이다.

$$R^i : \text{If } x_k \text{ is } A_{ki} \text{ then } Cy_{ki} = w_{ki} \quad (2)$$

$$R^i : \text{If } x_k \text{ is } A_{ki} \text{ then } Cy_{ki} = ws_{ki} + w_{ki} \cdot x_k \quad (3)$$

식(2)와 식(3)의 최종 추론결과는 보수적인 특성의 멤버쉽함수에 의해 동시에 두 개의 멤버쉽함수에만 영향을 받기 때문에 각각 식(4)와 식(5)가 된다. 이러한 구조로부터 기존의 NFN은 추론속도가 빠른 특징을 가지게 된다[13-15].

$$f_k(x_k) = \mu_{ki} \cdot w_{ki} + \mu_{ki+1} \cdot w_{ki+1} \quad (4)$$

$$f_k(x_k) = \mu_{ki} \cdot (ws_{ki} + w_{ki} \cdot x_k) + \mu_{ki+1} \cdot (ws_{ki+1} + w_{ki+1} \cdot x_k) \quad (5)$$

NFN의 학습은 그림 2에서 연결가중치를 모델에 가장 적절한 값으로 조정해 나가면서 이루어진다. 학습방법은 신경망의 학습에 가장 일반적으로 사용되는 오류역전파(Back-propagation; BP[2,3,13-15]) 알고리즘이 사용되며, 이는 최종 출력과 원 데이터간의 차, 즉 오차를 바탕으로 이루어진다. 연결가중치의 변화량은 간략 퍼지추론의 경우 식(6), 선형 퍼지추론의 경우 식(7)과 같다. 여기서 y 는 주어진 데이터, \hat{y} 는 모델의 출력, η 는 학습율, α 는 모멘텀계수이다. η 와 α 는 0과 1 사이의 값을 갖는다.

$$\Delta w_{ki}(t+1) = 2 \cdot \eta \cdot (y - \hat{y}) \cdot \mu_{ki} + \alpha \cdot \Delta w_{ki}(t) \quad (6)$$

$$\begin{cases} \Delta ws_{ki}(t+1) = 2 \cdot \eta \cdot (y - \hat{y}) \cdot \mu_{ki} + \alpha \cdot \Delta ws_{ki}(t) \\ \Delta w_{ki}(t+1) = 2 \cdot \eta \cdot (y - \hat{y}) \cdot \mu_{ki} \cdot x_k + \alpha \cdot \Delta w_{ki}(t) \end{cases} \quad (7)$$

2.2 뉴로퍼지 네트워크의 다항식 퍼지추론 구조

제안된 NFN의 구조는 기존의 NFN을 확장한 구조로 그림 2와 같다. 이 구조는 식(8)과 같은 퍼지규칙으로 표현된다. 식(8)의 퍼지규칙 후반부 구조를 살펴보면 선형 퍼지추론 구조에 제곱 항을 추가함으로써 후반부 구조를 확장하였다. 또한 그림 2에 보여지는 것과 같이 연결가중치에 의해 회귀다항식이 구성되며, 간략 퍼지추론(Type 0), 선형 퍼지추론(Type 1), 회귀다항식 퍼지추론(Type 2)을 모두 포함하게 된다. 따라서 다항식 차수(Type)의 선택만으로도 각각의 퍼지추론 구조를 표현할 수 있다.

$$R^i : \text{If } x_k \text{ is } A_{ki} \text{ then } Cy_{ki} = u0_{ki} + w1_{ki} \cdot x_k + u2_{ki} \cdot x_k^2 \quad (8)$$

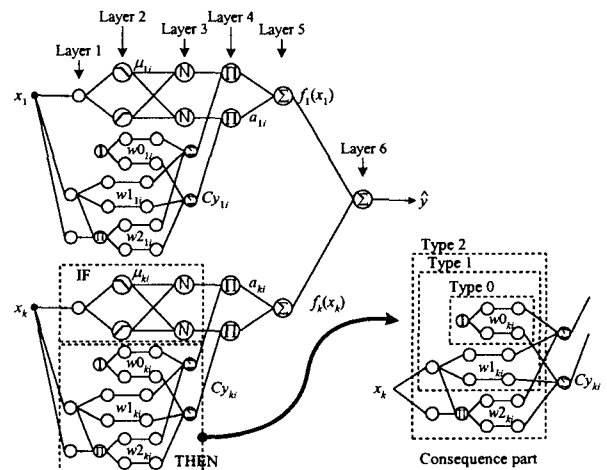


그림 2 NFN의 다항식 퍼지추론 구조
 Fig. 2 Topology of NFN with the polynomial fuzzy inference

제안된 NFN 구조의 설계과정은 다음과 같다.

[Layer 1] 입력층 : 각 뉴런은 입력변수와 대응하며 퍼지공간으로 입력신호를 직접 전달한다.

[Layer 2] 멤버쉽함수에 대한 적합도 계산 : 각 뉴런은 입력변수에 대해 정의된 퍼지집합, 즉 각 언어변수에 대응하며, 입력신호에 대한 멤버쉽함수(언어변수)의 적합도가 결정된다.

[Layer 3] 정규화된 적합도 계산 : 멤버쉽함수의 적합도가

구해지면, 각 규칙에 대한 정규화된 적합도 $\bar{\mu}_{ki}$ 를 계산한다. 입력변수 x_k 는 앞장에서 언급했듯이 두개의 멤버쉽함수에만 영향을 받기 때문에 $\bar{\mu}_{ki} = \mu_{ki}$ 가 된다[13-15].

[Layer 4] 연결 가중치와의 곱 : 3층에서 구해진 적합도는 연결가중치에 의한 후반부 출력과 곱해져 각 규칙의 출력을 생성하고 다음 층의 입력이 된다.

$$a_{ki} = \bar{\mu}_{ki} \times Cy_{ki} = \mu_{ki} \times Cy_{ki} \quad (9)$$

연결가중치에 의한 후반부 출력 Cy_{ki} 는 각 규칙의 다항식 차수(Type)에 따라 다음과 같이 표현된다.

$$\begin{cases} \text{Type 0: } Cy_{ki} = u0_{ki} \\ \text{Type 1: } Cy_{ki} = u0_{ki} + w1_{ki} \cdot x_k \\ \text{Type 2: } Cy_{ki} = u0_{ki} + w1_{ki} \cdot x_k + u2_{ki} \cdot x_k^2 \end{cases} \quad (10)$$

[Layer 5] 각 입력의 퍼지규칙에 대한 퍼지추론 : NFN의 5층에서 각 뉴런 출력은 무게중심법에 의한 퍼지추론에 의해 구해지며, 보수적인 멤버쉽함수의 연산특징에 의해 식(11)과 같이 정리된다.

$$\begin{aligned} f_k(x_k) &= \sum_{i=1}^n a_{ki} = \sum_{i=1}^n \bar{\mu}_{ki} \cdot Cy_{ki} = \sum_{i=1}^n \mu_{ki} \cdot Cy_{ki} \\ &= \frac{\mu_{ki} \cdot Cy_{ki} + \mu_{ki+1} \cdot Cy_{ki+1}}{\mu_{ki} + \mu_{ki+1}} \\ &= \mu_{ki} \cdot Cy_{ki} + \mu_{ki+1} \cdot Cy_{ki+1} = a_{ki} + a_{ki+1} \end{aligned} \quad (11)$$

[Layer 6] NFN의 출력 : 5층에서 출력된 값들은 식(1)과 같이 모두 대수적으로 합해져 NFN의 출력 \hat{y} 를 최종적으로 계산하게 된다.

제안된 NFN 구조는 퍼지규칙 후반부의 다항식 차수를 선택함으로써 적절한 모델을 설계할 수 있다. 뿐만 아니라 퍼지규칙의 후반부를 그림 3과 식(12)의 예와 같이 각각 다른 다항식 차수의 구조로 나타낼 수 있으며, 일률적인 형식의 구조를 벗어나 주어진 시스템의 특성에 따라 유연한 구조를 생성할 수 있도록 한다. 또한 NFN의 성능을 개선하면서 퍼지규칙의 후반부 구성을 간략화 시킬 수 있고, 파라미터의 수를 줄이게된다. 이것은 퍼지공간과 출력과의 관계를 찾아 출력에 영향을 미치는 퍼지공간에 대해 다항식의 차수를 높여주고, 그렇지 않으면 차수를 낮춰주는 특징을 가진다.

$$\begin{aligned} R^{11} : & \text{If } x_1 \text{ is } A_{11} \text{ then } Cy_{11} = u0_{11} \\ R^{12} : & \text{If } x_1 \text{ is } A_{12} \text{ then } Cy_{12} = u0_{12} + w1_{12} \cdot x_1 \\ R^{21} : & \text{If } x_k \text{ is } A_{k1} \text{ then } Cy_{k1} = u0_{k1} \\ R^{22} : & \text{If } x_k \text{ is } A_{k2} \text{ then } Cy_{k2} = u0_{k2} + w1_{k2} \cdot x_k + u2_{k2} \cdot x_k^2 \end{aligned} \quad (12)$$

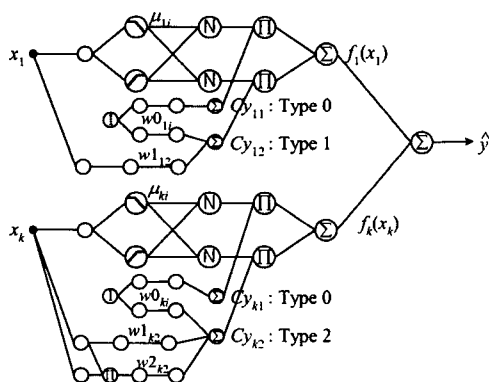


그림 3 후반부가 각 퍼지규칙에 대해 다르게 구성된 NFN
Fig. 3 NFN architecture; polynomials of consequence have different orders for each fuzzy rules

2.3 뉴로퍼지 네트워크의 학습

제안된 NFN 구조의 학습은 네트워크의 연결가중치 $w0_{ki}$, $w1_{ki}$, $w2_{ki}$ 를 모델에 가장 적절한 값으로 조정해 나가면서 이루어진다. 학습방법은 BP 알고리즘이 사용되며, 식(13)과 같은 출력오차를 이용한다. N개의 입출력 데이터 쌍이 주어진 경우, NFN의 최종 출력 오차는 식(14)와 같다.

$$E_p = (y_p - \hat{y}_p)^2 \quad (13)$$

$$E = \sum_{p=1}^N (y_p - \hat{y}_p)^2 \quad (14)$$

연결 가중치의 조절은 식(15)와 같이 기존의 값에 오차로부터 구해진 조절량 만큼의 변화를 더해주면 된다. 여기서 (t+1)은 새로운 값, (t)는 기존의 값을 의미한다. 연결가중치의 변화량 $\Delta w(t+1)$ 는 식(16)과 같이 구할 수 있다.

$$w(t+1) = w(t) + \Delta w(t+1) \quad (15)$$

$$\Delta w = \eta \left(- \frac{\partial E_p}{\partial w} \right) \quad (16)$$

식(16)으로부터 NFN의 연결가중치 변화량 및 그에 대한 사슬법칙(chain rule) 전개는 다음과 같다. 여기서, k는 k번째 입력변수, i는 i번째 퍼지규칙이다.

$$\Delta u0_{ki} = -\eta \frac{\partial E_p}{\partial u0_{ki}}, \quad \Delta w1_{ki} = -\eta \frac{\partial E_p}{\partial w1_{ki}}, \quad \Delta u2_{ki} = -\eta \frac{\partial E_p}{\partial u2_{ki}} \quad (17)$$

$$- \frac{\partial E_p}{\partial u0_{ki}} = - \frac{\partial E_p}{\partial \hat{y}_p} \frac{\partial \hat{y}_p}{\partial f_k(x_k)} \frac{\partial f_k(x_k)}{\partial a_{ki}} \frac{\partial a_{ki}}{\partial Cy_{ki}} \frac{\partial Cy_{ki}}{\partial u0_{ki}} \quad (18)$$

$$- \frac{\partial E_p}{\partial w1_{ki}} = - \frac{\partial E_p}{\partial \hat{y}_p} \frac{\partial \hat{y}_p}{\partial f_k(x_k)} \frac{\partial f_k(x_k)}{\partial a_{ki}} \frac{\partial a_{ki}}{\partial Cy_{ki}} \frac{\partial Cy_{ki}}{\partial w1_{ki}} \quad (19)$$

$$- \frac{\partial E_p}{\partial u2_{ki}} = - \frac{\partial E_p}{\partial \hat{y}_p} \frac{\partial \hat{y}_p}{\partial f_k(x_k)} \frac{\partial f_k(x_k)}{\partial a_{ki}} \frac{\partial a_{ki}}{\partial Cy_{ki}} \frac{\partial Cy_{ki}}{\partial u2_{ki}} \quad (20)$$

식(17)~식(20)의 공통항에 대한 유도과정은 다음과 같다.

$$- \frac{\partial E_p}{\partial \hat{y}_p} = - \frac{\partial}{\partial \hat{y}_p} (y_p - \hat{y}_p)^2 = 2 \cdot (y_p - \hat{y}_p) \quad (21)$$

$$\frac{\partial \hat{y}_p}{\partial f_k(x_k)} = \frac{\partial}{\partial f_k(x_k)} \sum_k f_k(x_k) = 1 \quad (22)$$

$$\frac{\partial f_k(x_k)}{\partial a_{ki}} = \frac{\partial}{\partial a_{ki}} (a_{ki} + a_{ki+1}) = 1 \quad (23)$$

$$\frac{\partial a_{ki}}{\partial Cy_{ki}} = \frac{\partial}{\partial Cy_{ki}} (\mu_{ki} \cdot Cy_{ki}) = \mu_{ki} \quad (24)$$

식(18)~식(20)의 마지막 부분에 대한 유도과정은 퍼지규칙의 다항식 차수에 따라 달라진다.

i) Type 0 : $Cy_{ki} = u0_{ki}$

$$\frac{\partial Cy_{ki}}{\partial u0_{ki}} = \frac{\partial u0_{ki}}{\partial u0_{ki}} = 1 \quad (25)$$

ii) Type 1 : $Cy_{ki} = u0_{ki} + w1_{ki} \cdot x_k$

$$\begin{cases} \frac{\partial Cy_{ki}}{\partial u0_{ki}} = \frac{\partial}{\partial u0_{ki}} (u0_{ki} + w1_{ki} \cdot x_k) = 1 \\ \frac{\partial Cy_{ki}}{\partial w1_{ki}} = \frac{\partial}{\partial w1_{ki}} (u0_{ki} + w1_{ki} \cdot x_k) = x_k \end{cases} \quad (26)$$

iii) Type 2 : $Cy_{ki} = u0_{ki} + w1_{ki} \cdot x_k + u2_{ki} \cdot x_k^2$

$$\begin{cases} \frac{\partial Cy_{ki}}{\partial u0_{ki}} = \frac{\partial}{\partial u0_{ki}} (u0_{ki} + w1_{ki} \cdot x_k + u2_{ki} \cdot x_k^2) = 1 \\ \frac{\partial Cy_{ki}}{\partial w1_{ki}} = \frac{\partial}{\partial w1_{ki}} (u0_{ki} + w1_{ki} \cdot x_k + u2_{ki} \cdot x_k^2) = x_k \\ \frac{\partial Cy_{ki}}{\partial u2_{ki}} = \frac{\partial}{\partial u2_{ki}} (u0_{ki} + w1_{ki} \cdot x_k + u2_{ki} \cdot x_k^2) = x_k^2 \end{cases} \quad (27)$$

식(15)~식(27)로부터 모멘텀이 포함된 연결가중치 변화량은 다항식 차수에 따라 다음과 같이 정의된다.

i) Type 0 : $Cy_{ki} = w0_{ki}$

$$\Delta w0_{ki}(t+1) = 2 \cdot \eta \cdot (y_p - \hat{y}_p) \cdot \mu_{ki} + \alpha \cdot \Delta w0_{ki}(t) \quad (28)$$

ii) Type 1 : $Cy_{ki} = w0_{ki} + w1_{ki} \cdot x_k$

$$\begin{cases} \Delta w0_{ki}(t+1) = 2 \cdot \eta \cdot (y_p - \hat{y}_p) \cdot \mu_{ki} + \alpha \cdot \Delta w0_{ki}(t) \\ \Delta w1_{ki}(t+1) = 2 \cdot \eta \cdot (y_p - \hat{y}_p) \cdot \mu_{ki} \cdot x_k + \alpha \cdot \Delta w1_{ki}(t) \end{cases} \quad (29)$$

iii) Type 2 : $Cy_{ki} = w0_{ki} + w1_{ki} \cdot x_k + w2_{ki} \cdot x_k^2$

$$\begin{cases} \Delta w0_{ki}(t+1) = 2 \cdot \eta \cdot (y_p - \hat{y}_p) \cdot \mu_{ki} + \alpha \cdot \Delta w0_{ki}(t) \\ \Delta w1_{ki}(t+1) = 2 \cdot \eta \cdot (y_p - \hat{y}_p) \cdot \mu_{ki} \cdot x_k + \alpha \cdot \Delta w1_{ki}(t) \\ \Delta w2_{ki}(t+1) = 2 \cdot \eta \cdot (y_p - \hat{y}_p) \cdot \mu_{ki} \cdot x_k^2 + \alpha \cdot \Delta w2_{ki}(t) \end{cases} \quad (30)$$

여기서, $\Delta w0_{ki}(t) = w0_{ki}(t) - w0_{ki}(t-1)$.

3. 진화론적 최적 뉴로퍼지 네트워크

이 장에서는 GAs 기반한 진화론적 최적 뉴로퍼지 네트워크(GoNFN)의 구조와 광범위한 설계방법론을 소개한다. 이를 위해 동적 탐색 기반 GAs를 제안하며, 진화과정을 통해 NFN의 최적 구조인 GoNFN을 설계한다.

3.1 동적 탐색 기반 유전자 알고리즘

유전자 알고리즘(GAs[7,8,21])은 “적자생존”의 생물학 원리에 바탕을 둔 최적화 기법중의 하나로 자연계의 생명체 중 환경에 잘 적응한 개체가 좀더 많은 자손을 남길 수 있다는 자연선택 과정과 생명체의 설계도와 같은 유전자의 변화를 통해서 좋은 방향으로 발전해 나간다는 자연진화의 과정인 유전자 진화 메커니즘에 바탕을 둔 탐색 알고리즘이다. 이는 실세계의 문제를 풀기 위해 잠재적인 해들을 컴퓨터 상에서 코딩된 개체로 나타내고, 여러 개의 개체들을 모아 군집을 형성한 뒤, 세대를 거듭하면서 이들의 유전 정보를 서로 교환하거나 새로운 유전 정보를 부여하여 적자 생존의 법칙에 따른 진화과정을 통해, 주어진 문제에 대한 최적의 해를 찾아가는 방법이다[7,8].

GAs를 이용한 최적화 과정 중 탐색이 침체되는 원인은 몇 가지 있으며, 그중 한가지 원인은 전역해가 존재하는 해 공간에 대한 정의이다. 대상 시스템에 대한 전역해의 탐색은 변수의 부호화와 복호화 과정을 통해 이루어진다. 부호화 과정에서 중요한 파라미터는 변수에 할당된 문자열의 길이, 즉 이진 문자열의 비트 수이며, 복호화 과정에서 중요한 파라미터는 해 공간에서 정의된 변수 x의 범위 $[x_{low}, x_{high}]$ 이다. 여기서 문자열의 길이는 주어진 탐색공간 내의 점들을 기술할 수 있도록 선택되어야 하며, 범위 역시 전역 해를 포함하도록 정의되어야 한다. 일반적으로 탐색공간은 대상 시스템에 대해 정의되며, 주어진 탐색공간으로부터 문자열의 길이를 선택하고, 이를 바탕으로 최적해를 탐색한다. 그러나 탐색범위가 크게 주어지게 되면, 문자열은 주어진 범위에 비례하여 지수적으로 커지게 되고 많은 연산시간의 요구, 더딘 해의 탐색 등 GAs의 성능을 저하시키는 요인이 된다. 반대로 탐색범위가 작게 주어지게 되면, 문자열의 길이는 짧아지고 연산부담은 적어지지만 해의 정밀도는 떨어지게 된다.

따라서 본 논문에서는 이러한 요인을 개선하기 위해 동적인 탐색공간의 정의를 제안한다. 이 방법은 충분히 큰 탐색

공간에서 해의 영역을 탐색한 후 기준 해를 정의하고 기준 해의 이동 범위를 바탕으로 탐색영역을 조절해 가며 최적의 해를 찾아간다. 그림 4는 동적 탐색 기반 GAs의 탐색공간의 조절과정을 보여주고 있으며, 간단한 예를 들어 단계별로 살펴보면 다음과 같다.

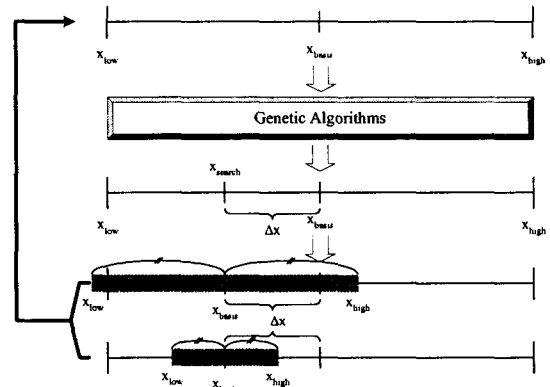


그림 4 동적 탐색 기반 GAs
Fig. 4 Dynamic search based GAs

[단계 1] 주어진 변수 x에 대해 탐색공간 $[x_{low}, x_{high}]$ 과 탐색공간에 대한 적절한 문자열의 길이 그리고 기준 해를 정의한다. 예를 들어 x에 대해 탐색영역은 $[0, 100]$, 문자열의 길이는 10비트, 기준해는 50이라 가정하자. 기준해는 미리 알고 있는 임의의 값이나, 탐색영역의 중심으로 정의한다.

[단계 2] GAs를 이용하여 $[0, 100]$ 에 대해 해를 탐색한다.

[단계 3] 충분한 탐색 후, 즉 최적해가 세대를 거듭해도 변하지 않게 되면, 미리 정해진 기준 해와 GAs에 의해 선택된 해를 비교한다. GAs에 의해 선택된 해를 기준 해로 다시 정의하고 기준 해의 이동에 비례하여 탐색공간을 재 정의한다.

$$\Delta x = x_{basis} - x_{search} \quad (31)$$

$$x_{basis} = x_{search} \quad (32)$$

$$x_{range} = [x_{basis} - (\epsilon \cdot \Delta x), x_{basis} + (\epsilon \cdot \Delta x)] \quad (33)$$

여기서, x_{basis} 는 기준 해, x_{search} 는 선택된 해, Δx 는 기준 해와 선택된 해의 변화량, 즉 기준 해의 이동을 나타낸다. ϵ 은 기준 해의 이동에 비례하여 적절하게 조절함으로써, 탐색 범위를 결정하도록 한다. 예를 들어 만약 GAs가 10.6이라는 해를 선택하였다고 가정하자. 새로운 기준 해는 10.6이 되며, 최적 해는 x_{low} 방향으로 39.4만큼 이동하게 된다. 이를 기준으로 탐색공간을 $[-28.8, 50]$ 으로 재조정한다. 이때 $\epsilon = 1$ 이다.

[단계 4] 새로운 영역에 대해 최적 해를 탐색하며 단계 2, 단계 3을 반복한다. 여기서 기준 해의 변화를 계속 확인하며, 변화에 비례하여 적절한 ϵ 을 선택한다. 즉, 변화량이 클 경우 $\epsilon \geq 1$ 을 선택하고, 변화가 작은 경우 $\epsilon < 1$ 을 선택한다. 변화가 크다는 의미는 설정된 범위가 잘못되었음을 의미하며, 그림 4의 예와 같이 탐색공간으로 정의되지 않았던 공간으로 범위를 재조정해야 할 필요가 있기 때문이다. 기준 해의 변화량이 작은 경우는 변수의 값이 최적 해의 근처에서 머물고 있는 상태이며, 조금 더 자세한 정보의 탐색을 위해 탐색공간을 축소한다. 본 논문에서 제안된 탐색공간의 재 정의는 문자열 길이의 변화 없이 해에 대한 정밀도를 높이는 효과가 있다. 예를 들어 기준 해가 다시 8.5에서 정의되었다 하자. ϵ

=1일 때, 식(45)~식(47)로부터 $x_{range}=[8.5-2.1, 8.5+2.1]=[7.4, 10.6]$ 이 되며, 처음 정의되었던 탐색공간 $[0, 100]$ 에 대해 10비트의 문자열은 $(100-0)/1023=0.0978$ 의 간격으로 공간을 탐색하지만, 재 정의된 탐색공간에 대해서는 $(10.6-7.4)/1023=0.0031$ 까지 정밀도를 높게 된다.

[단계 5] 변화하지 않거나, 만족하는 해를 얻게 되면, 알고리즘을 종료한다.

제안된 동적 탐색 기반 GAs의 특징은 다음과 같다.

- 1) 전역해가 존재하는 적절한 해의 공간을 제공한다.
- 2) 탐색공간에 대한 문자열의 부담을 덜어준다.
- 3) 잘못 정의된 탐색공간을 적절한 공간으로 재조정한다.
- 4) 문자열 길이의 변화 없이 해에 대한 정밀도를 높여준다.

3.2 진화론적 최적화

주어진 비선형 시스템에 대한 최적의 모델을 결정하기 위해서는 크게 두 가지 측면이 있다. 즉, 구조 동정과 파라미터 동정이다. 구조 동정은 입력변수의 선택, 공간분할, 퍼지규칙의 수 등 모델의 구조와 관계되며, 파라미터 동정은 멤버쉽함수의 파라미터, 후반부 다항식의 계수 등과 같이 모델의 세부값들을 정의하는 문제이다. 그러나 구조 문제의 경우 고려해야 할 사항들이 많기 때문에 동정과정의 복잡하고 만족스러운 결과를 얻지 못하였다. 따라서 대부분의 경우 구조를 정의하고, 그 구조에 대한 파라미터 결정방법에 대해 많은 부분이 연구되고 있다[4-6,14].

기존에 다루진 GAs를 이용한 최적화 문제는 대부분 파라미터에 관한 연구이며, 일부에서 구조 결정을 위해 GAs를 다루고 있지만 멤버쉽함수 수의 결정, 공간분할 등 일부부분에 국한되고 있다. 또한 구조를 다루더라도 파라미터와 별개로 다루어진다. 다시 말하면, 먼저 구조를 동조하고, 정의된 구조에 대한 파라미터를 다시 동조하는 방법을 이용하고 있다[15,22]. 그러나 복잡하고 다변수로 주어지는 비선형함수의 경우 이 방법은 전역 해를 찾지 못하고 지역 해에 빠질 수 있는 위험이 있으며, 구조와 파라미터의 잘못된 관계, 주어진 시스템을 고려하지 않은 일률적인 구조의 적용 등의 문제들이 발생된다.

표 1 GoNFN을 위해 GAs에서 다루어지는 문제
Table 1 Problems handled by GAs for GoNFN

	○	·	·	○
	Tuned	·	○	Tuned
	○	·	○	○
	≤k ○	·	·	○
	Tuned	·	○	Tuned
	≤k ○	·	○	○

본 논문에서는 이러한 문제들을 해결하기 위해 GAs를 이용하여 구조와 파라미터 문제를 다루고자 한다. 이를 위해, NFN 구조와 파라미터 동정에 대한 효율적인 방법을 논의한다. 표 1은 GoNFN 생성을 위해 다루어지는 문제들을 보여준다. 여기서 '○'는 GAs에 의해 동조됨을 의미하며, 'Tuned'은

동조된 값, '·'는 고정된 값의 이용을 의미한다. 'MF'는 멤버쉽함수, 'Para.'는 파라미터, 'Order'는 후반부 다항식의 차수를 나타낸다. GAs를 이용한 NFN의 최적화는 2가지 방향에서 이루어진다. 첫 번째는 NFN의 구조를 동조한 후, 파라미터를 동조하는 방법이고(㉔), 두 번째는 구조 및 파라미터를 동시에 동조하는 방법이다(㉕).

첫 번째 방법에 의한 ㉔는 멤버쉽함수의 파라미터를 입력 데이터의 최소-최대값(Min-Max)으로 고정하고 최적의 구조를 탐색한 후, 동정된 구조에 다시 GAs를 적용하여 파라미터를 동조하는 방법이다. 이 방법은 적은 비트의 염색체, 빠른 탐색 등의 특징을 가지게 되지만, GAs를 두 번 적용해야 하고, 구조가 복잡해질 수 문제가 있다.

두 번째 방법인 ㉕는 첫 번째 방법과 달리 구조와 파라미터를 동시에 동조한다. 이 방법은 구조와 파라미터를 모두 고려해야 하기 때문에 많은 비트의 염색체 이용과 많은 탐색 시간의 소요 등의 문제를 가지고 있지만, 최적 해에 대한 탐색 결과는 오히려 좋은 결과를 얻을 수 있다. 뿐만 아니라, 구조 동정 후 파라미터를 재 동조해야 할 필요가 없기 때문에 한번의 GAs 적용으로도 원하는 모델을 구축할 수 있다. ㉔k와 ㉕k는 ㉔와 ㉕에 전반부 입력의 수를 제한한 경우이다. 즉 주어진 전체 입력 n개의 입력변수 중 k보다 적은 수의 입력변수를 선택하도록 제한하였다. 시스템의 입력변수가 많이 주어지는 경우 퍼지규칙의 수가 증가하고, 학습 데이터에 대한 모델 성능, 즉 근사화에 대해서는 우수한 결과를 얻을 수 있으나, 테스트 데이터에 대한 결과, 즉 일반화에 대한 성능은 현저히 저하된다. 따라서 이러한 문제를 해결하기 위해 입력 수를 제한한다.

시스템 입력의 동정문제는 주어진 전체 입력으로부터 모델 출력에 영향을 미치는 입력을 결정하는 것이다. 물론 주어진 시스템 입력 모두를 선택할 때 좋은 결과를 얻을 수 있겠지만, 많은 입력중 주요 변수들을 결정하기 어려울 때, 또는 시스템 입력이 많이 주어지게 되면 규칙 수와 파라미터 수의 증가, 많은 연산시간 및 메모리 요구 등, 성능향상에 비해 오히려 더 많은 문제를 발생시킬 수 있다. 따라서 모델에 적절한 입력변수의 선택은 중요한 문제이다. 두 번째로 퍼지공간의 분할과 관계되는 문제로 멤버쉽함수(MF)의 수를 결정하는데 있다. 퍼지공간의 분할은 입력변수와 함께 퍼지규칙 수를 결정하기 때문에 멤버쉽함수 수의 증가는 입력변수의 증가와 같은 문제를 야기하게 된다. 따라서 각 입력에 대해 적절한 멤버쉽함수 수의 선택이 이루어져야 한다. 다음은 후반부 구조 문제이다. 후반부는 전반부 퍼지공간의 표현과 관련되며, 분할되어진 각각의 퍼지공간이 어떠한 형태로 나타내어지는가를 결정해야 한다. NFN의 파라미터 문제는 전반부 멤버쉽함수의 파라미터와 후반부 다항식의 계수를 결정하는 것이다. 후반부 파라미터의 경우, BP에 의한 학습에 의해 이루어지며, 멤버쉽함수의 파라미터는 GAs를 이용한다.

본 논문에서는 위에서 제시한 방법(㉔~㉕k)으로부터 각 경우에 대한 GoNFN을 생성하며, 성능 및 구조의 비교를 통해 효율적인 방법이 무엇인지에 대해 논의한다.

3.3 GoNFN의 설계

진화론적 최적 NFN(GoNFN) 구조는 앞 절에서 소개한 NFN 구조의 최적 구조로 진화과정을 통해 주어진 문제에 대

해 유연한 구조를 가지게 된다. 그림 5는 GoNFN 구조를 위한 전체 흐름도를 보여준다.

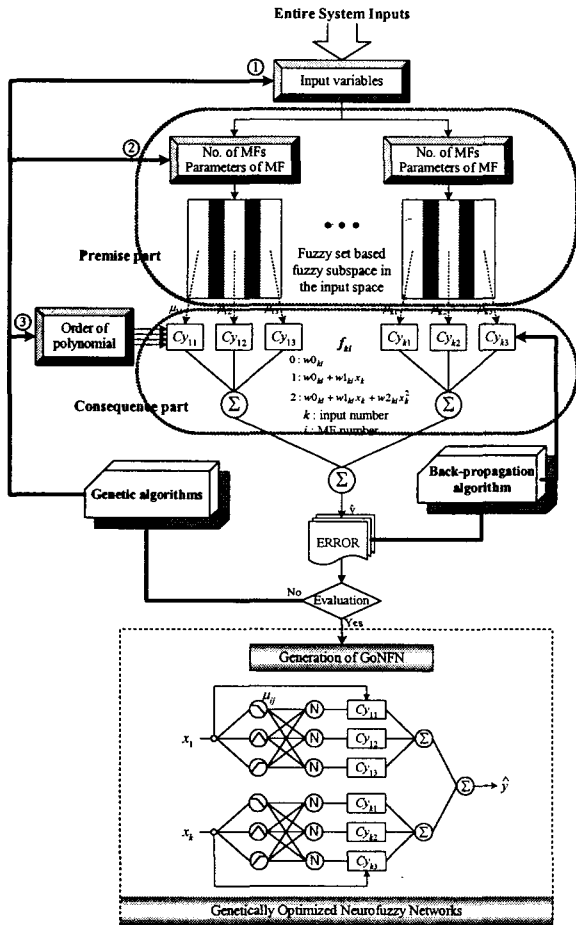


그림 5 GoNFN 구조의 설계과정
Fig. 5 Design procedure of GoNFN architecture

[단계 1] 주어진 전체 시스템 입력에 대해 GAs의 탐색체 정보로부터 모델의 입력변수를 선택한다(①). 선택된 입력을 퍼지공간으로 전달한다.

[단계 2] 선택된 입력의 퍼지공간을 정의하기 위한 탐색체 정보를 가져온다(②). 탐색체는 각 입력변수에 대해 정의된 멤버쉽함수의 수와 파라미터에 대한 정보를 가진다. 유전자 정보에 의해 분할된 퍼지공간으로부터 퍼지규칙의 적합도를 결정한다.

[단계 3] NFN 후반부 구조를 결정한다(③). 후반부 구조는 그림 3과 같이 선택된 입력과 각각의 퍼지규칙에 대해 다르게 정의된다. 연결가중치에 의한 네트워크 구조 역시 선택된 후반부 구조에 따라 다른 형태로 나타나게 된다. 표현의 간결성을 위해 후반부 네트워크 구조를 사각형의 블록으로 표현하였다.

[단계 4] 각 퍼지규칙의 적합도는 연결가중치에 의한 후반부 출력과 곱해져 규칙의 출력을 생성하고, 회귀다항식 퍼지추론에 의해 입력에 대한 출력을 식(12)과 같이 계산한다. 식(1)로부터 NFN의 출력 \hat{y} 을 구한다.

[단계 5] NFN의 최종출력과 실제 출력으로부터 오차를 구한다. 구하여진 오차는 BP 알고리즘을 통해 네트워크의 역방향으로 전달되며, 이를 바탕으로 연결가중치 w_{0k} , w_{1k} , w_{2k} 를

조정하는 학습과정이 이루어진다.

[단계 6] 학습이 종료된 NFN 구조는 탐색체의 진화를 위해 평가된다.

[단계 7] NFN의 정보를 가지고 있는 각 유전자 개체는 적합도(fitness)와 함께 최적 해 탐색을 위한 GAs의 정보가 되며, 최적의 모델이 생성될 때까지 유전자의 탐색체는 [단계 1]에서 [단계 6]을 반복하며 진화한다. 마지막 세대까지 진화한 개체는 최적의 NFN 모델에 대한 정보를 가지게 된다. 즉 최적의 개체는 GoNFN을 생성한다.

4. 실험연구 및 결과고찰

GoNFN 구조의 생성 및 평가를 위해 NOx 배출공정과 가스 공정에 적용한다. GAs에 의해 생성된 GoNFN은 각 공정에 대해 유연성을 가지며, 각각 다른 구조의 형태로 나타난다. GoNFN 구조 동정에 이용되는 GAs의 연산자와 파라미터는 표 2와 같다. 여기서 문자열의 길이는 앞 절에서 제시한 GAs의 동조 방법에 따라 다르게 설정된다.

표 2 유전자 알고리즘의 연산자와 파라미터

Table 2 Operators and parameters of genetic algorithms

	1000
	60
	Roulette-wheel selection
	One-point crossover
	0.75
	invert mutation
	0.065
	Set according to a case

4.1 가스터빈 발전소의 NOx 배출공정

대기상 질소산화물(NOx)은 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(질소 산화물)로 일컫는다.

NOx의 저감방법으로는 발생원으로부터 NOx의 배출을 억제하는 방법과 배출가스중의 NOx를 제거하는 방법을 들 수 있는데, 최근에는 Gas Turbine에 사용하는 입출력 변수 중에서, 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[16].

$$PI = \frac{1}{N} \sum_{p=1}^N (y_p - \hat{y}_p)^2 \tag{34}$$

본 논문에서는 이러한 입출력 변수들을 이용하여 제안된 모델의 성능을 평가한다. 가스터빈 발전소의 NOx 배출 공정은 5입력 1출력 [Tamb,COM,LPT,Pcd,Texh;NOx]으로, [x₁,x₂,

$x_3, x_4, x_5; y]$ 로 표기하며, NO_x의 배출 메카니즘을 모델링하고, GoNFN 모델을 구축한다. 제안된 모델의 성능을 테스트하기 위해서 전체 260개의 데이터를 130개씩 두 개의 그룹으로 랜덤하게 선택하여 각각 학습데이터와 테스트 데이터로 사용하며, 모델의 근사화와 일반화 성능을 얻는다. 근사화(PI)와 일반화(E_PI) 능력을 측정하는 성능지수는 식(34)와 같다.

표 3 NO_x 배출공정에 대한 GoNFN의 성능지수

Table 3 Performance index of GoNFN for the NO_x emission process

Case	Input Variables	Structure	Method	BP	PI	E_PI
A	GAs	2+2+2+2	Min-Max	BP	7.918	10.676
	Tuned (x ₁ , x ₂ , x ₃ , x ₄ , x ₅)	2+2+2+2	GAs	BP	7.338	9.666
B	GAs	2+2+2+2	GAs	BP	7.972	10.216
A ₄	GAs	2+2+2+2	Min-Max	BP	10.158	13.141
	Tuned (x ₁ , x ₂ , x ₃ , x ₅)	2+2+2+2	GAs	BP	8.823	11.197
B ₄	GAs	2+2+2	GAs	BP	8.474	10.494
A ₃	GAs	2+2+2	Min-Max	BP	16.003	15.837
	Tuned (x ₁ , x ₃ , x ₅)	2+2+2	GAs	BP	14.522	13.865
B ₃	GAs	2+2+2	GAs	BP	9.774	10.002
B	GAs	6+6+6+6	GAs	BP	3.453	3.973
B	x ₁ , x ₂ , x ₃ , x ₄ , x ₅	6+6+6+6+6	GAs	BP	3.286	4.067

표 3은 본 논문에서 제안된 방법들에 대한 모델의 성능지수를 나타낸다. A와 B는 앞 절에서 언급했던 것과 같이 A는 입력변수와 다항식차수 동조 후 멤버쉽함수의 파라미터를 동조한 결과이며, B는 GAs를 이용하여 입력변수, 멤버쉽함수의 파라미터, 규칙 후반부의 다항식차수를 동시에 동조하는 방법이다. 멤버쉽함수의 수는 네트워크의 구조, 퍼지규칙의 수, 퍼지공간의 분할 등을 고려하여 각 입력에 대해 2개를 사용한다. A₄와 A₃은 A의 방법에 입력변수의 수를 제한하였다. 즉, 표 3에서 보여지는 것처럼 A₄는 4입력 이하의 입력변수를 선택하도록 하였고, A₃은 최대 3입력을 선택하도록 하였다. B의 방법 역시 입력변수를 제한하여 결과를 비교하였다. 표 3의 결과로부터 알 수 있듯이 B를 이용한 방법, 즉, 구조 동조와 파라미터 동조를 포함한 동조방법이 효율적이며, 좋은 성능의 결과를 얻을 수 있었다. 뿐만 아니라, 입력변수의 수를 제한하였을 경우 A의 방법은 입력 수의 제한조건에 따라 출력특성이 많이 달라지는 반면 B의 방법은 제한조건에 따른 변화가 심하게 나타나지 않고 있다.

식(35)와 식(36)은 A와 B의 퍼지규칙을, 식(37)과 식(38)은 A₃과 B₃의 퍼지규칙을 비교하여 보여주고 있다. 식(35)는 26개의 후반부 파라미터, 식(36)은 22개의 후반부 파라미터를 가진다. 그러나 출력특성은 큰 차이가 없으며, 오히려, 식(35)는 GAs를 두 번 이용해야만 하기 때문에 GoNFN 구조는 B로부터 얻어진다. 3보다 작은 입력선택의 조건으로부터 식(37)과 식(38)의 퍼지규칙은 생성된다. 이 두 퍼지규칙은 유사한 구조를 가지지만 성능면에서 큰 차이를 보인다. B₃의 방법에 의해 생성된 식(38)의 구조가 좋은 출력특성을 나타내며, 후반부 파라미터 역시 1개 적은 구조를 가진다.

$$\begin{aligned}
 R_{11} &: \text{If } x_1 \text{ is } A_{11} \text{ then } y_{11} = u_{011} \\
 R_{12} &: \text{If } x_1 \text{ is } A_{12} \text{ then } y_{12} = u_{012} + w_{112}x_1 + u_{212}x_1^2 \\
 R_{21} &: \text{If } x_2 \text{ is } A_{21} \text{ then } y_{21} = u_{021} + w_{121}x_2 + u_{221}x_2^2 \\
 R_{22} &: \text{If } x_2 \text{ is } A_{22} \text{ then } y_{22} = u_{022} + w_{122}x_2 \\
 R_{31} &: \text{If } x_3 \text{ is } A_{31} \text{ then } y_{31} = u_{031} + w_{131}x_3 + u_{231}x_3^2 \\
 R_{32} &: \text{If } x_3 \text{ is } A_{32} \text{ then } y_{32} = u_{032} + w_{132}x_3 + u_{232}x_3^2 \\
 R_{41} &: \text{If } x_4 \text{ is } A_{41} \text{ then } y_{41} = u_{041} + w_{141}x_4 + u_{241}x_4^2 \\
 R_{42} &: \text{If } x_4 \text{ is } A_{42} \text{ then } y_{42} = u_{042} + w_{142}x_4 + u_{242}x_4^2 \\
 R_{51} &: \text{If } x_5 \text{ is } A_{51} \text{ then } y_{51} = u_{051} + w_{151}x_5 \\
 R_{52} &: \text{If } x_5 \text{ is } A_{52} \text{ then } y_{52} = u_{052} + w_{152}x_5 + u_{252}x_5^2
 \end{aligned} \tag{35}$$

$$\begin{aligned}
 R_{11} &: \text{If } x_1 \text{ is } A_{11} \text{ then } y_{11} = u_{011} \\
 R_{12} &: \text{If } x_1 \text{ is } A_{12} \text{ then } y_{12} = u_{012} + w_{112}x_1 + u_{212}x_1^2 \\
 R_{21} &: \text{If } x_2 \text{ is } A_{21} \text{ then } y_{21} = u_{021} + w_{121}x_2 \\
 R_{22} &: \text{If } x_2 \text{ is } A_{22} \text{ then } y_{22} = u_{022} \\
 R_{31} &: \text{If } x_3 \text{ is } A_{31} \text{ then } y_{31} = u_{031} + w_{131}x_3 + u_{231}x_3^2 \\
 R_{32} &: \text{If } x_3 \text{ is } A_{32} \text{ then } y_{32} = u_{032} + w_{132}x_3 \\
 R_{41} &: \text{If } x_4 \text{ is } A_{41} \text{ then } y_{41} = u_{041} + w_{141}x_4 \\
 R_{42} &: \text{If } x_4 \text{ is } A_{42} \text{ then } y_{42} = u_{042} + w_{142}x_4 + u_{242}x_4^2 \\
 R_{51} &: \text{If } x_5 \text{ is } A_{51} \text{ then } y_{51} = u_{051} + w_{151}x_5 \\
 R_{52} &: \text{If } x_5 \text{ is } A_{52} \text{ then } y_{52} = u_{052} + w_{152}x_5 + u_{252}x_5^2
 \end{aligned} \tag{36}$$

$$\begin{aligned}
 R_{11} &: \text{If } x_1 \text{ is } A_{11} \text{ then } y_{11} = u_{011} + w_{111}x_1 \\
 R_{12} &: \text{If } x_1 \text{ is } A_{12} \text{ then } y_{12} = u_{012} \\
 R_{21} &: \text{If } x_3 \text{ is } A_{21} \text{ then } y_{21} = u_{021} + w_{121}x_3 + u_{221}x_3^2 \\
 R_{22} &: \text{If } x_3 \text{ is } A_{22} \text{ then } y_{22} = u_{022} + w_{122}x_3 + u_{222}x_3^2 \\
 R_{31} &: \text{If } x_5 \text{ is } A_{31} \text{ then } y_{31} = u_{031} + w_{131}x_5 + u_{231}x_5^2 \\
 R_{32} &: \text{If } x_5 \text{ is } A_{32} \text{ then } y_{32} = u_{032}
 \end{aligned} \tag{37}$$

$$\begin{aligned}
 R_{11} &: \text{If } x_3 \text{ is } A_{11} \text{ then } y_{11} = u_{011} \\
 R_{12} &: \text{If } x_3 \text{ is } A_{12} \text{ then } y_{12} = u_{012} + w_{112}x_3 \\
 R_{21} &: \text{If } x_4 \text{ is } A_{21} \text{ then } y_{21} = u_{021} + w_{121}x_4 + u_{221}x_4^2 \\
 R_{22} &: \text{If } x_4 \text{ is } A_{22} \text{ then } y_{22} = u_{022} + w_{122}x_4 \\
 R_{31} &: \text{If } x_5 \text{ is } A_{31} \text{ then } y_{31} = u_{031} + w_{131}x_5 + u_{231}x_5^2 \\
 R_{32} &: \text{If } x_5 \text{ is } A_{32} \text{ then } y_{32} = u_{032}
 \end{aligned} \tag{38}$$

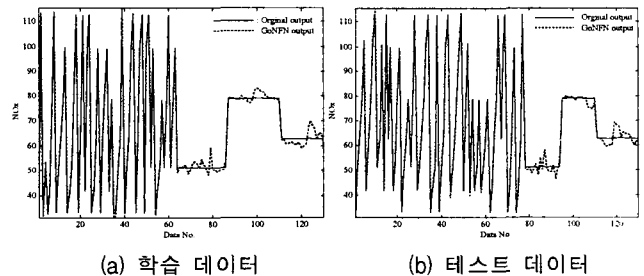


그림 6 NO_x 배출공정에 대한 GoNFN의 출력

Fig. 6 Output of GoNFN for the NO_x emission process

표 3은 각 입력에 대해 6개의 멤버쉽함수를 가지는 모델에 대한 결과를 포함하고 있다. 4입력의 GoNFN은 전체 입력 중 GAs에 의해 4개의 변수를 선택하였고, 이때 PI=3.453, E_PI=3.973의 성능지수를 가진다. 5입력의 GoNFN은 4입력의 GoNFN과 비교를 위해 입력변수를 동조하지 않고, 전반부 파라미터와, 후반부 구조를 동시에 동조하였다. 결과에서 보여지는 것과 같이 PI=3.286, E_PI =4.067의 성능을 가진다. 이 두 구조는 비슷한 결과를 보여주고 있지만, 퍼지규칙과 입력변수의 수에서 차이를 보인다. 즉 GAs로부터 시스템의 출력에 영향을 미치는 적절한 입력변수, 구조, 파라미터들이 선택된다. 그림 6은 24개의 퍼지규칙을 가지는 4입력 GoNFN 모델의 출력을 원출력과 비교하여 보여주고 있다.

표 4는 가스터빈 발전소의 NO_x 배출공정에 적용된 기존의 모델들과 성능지수를 비교하여 보여준다. Ahn의 모델은 전체 데이터를 이용하여 PI만을 고려한 값이다. FNN은 GAs

와 개선된 complex 알고리즘의 결합에 의한 하이브리드 알고리즘을 이용하여, 멤버쉽수의 파라미터를 동조한 구조이며, Multi-FNN은 클러스터를 이용한 구조로 4개의 FNN으로 구성된다. 즉 Multi-FNN의 구조를 위해 120의 퍼지규칙이 사용된다. 제안된 GoNFN은 GAs에 의한 구조 및 파라미터 동조로 적은 퍼지규칙 수에서도 좀 더 효율적이고, 객관적인, 그리고 대상 공정에 대해 적용할 수 있는 유연한 구조를 설계할 수 있도록 해준다.

표 4 기존 모델방법과의 동정오차 비교

Table 4 Comparison of performance with other modeling methods

Regression model		17.68	19.23	
Ahn's[24]	FNN	5.835		
	AIM	8.420		
FNN[14]	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)×4
	Linear	0.720	2.025	30(6+6+6+6+6)×4

4.2 가스로 시계열 공정

Box와 Jenkin[17]이 사용한 가스로 시계열 데이터를 이용하여 입출력 데이터인 가스 흐름을 $u(t)$ 와 연소된 이산화탄소 농도 $y(t)$ 를 모델 동정에 이용한다. 가스로 시계열 데이터는 모델링 기법의 성능을 평가하는데 널리 사용되는 표준데이터로써, 제안된 모델의 타당성과 유용성을 평가한다. 가스로 시계열 데이터는 입출력 데이터인 가스 흐름율과 연소된 이산화탄소 농도의 가스로 공정으로부터 추출된다. 추출된 데이터는 296개의 쌍으로 이루어지며, 모델을 구성하기 위해 시스템 입력을 과거의 시간에 따라 적절하게 선택하여 사용한다. 시스템의 입력변수로 기존의 모델링에 가장 많이 사용되는 $u(t-3), u(t-2), u(t-1), y(t-3), y(t-2), y(t-1)$ 의 6입력을 적용하며, 이들 중 GAs에 의한 입력선택을 통해 주어진 시스템에 적절한 입력변수를 동조한다. 뿐만 아니라, 입력변수 수를 제한하여 모델을 설계한다. 입력변수의 수는 다른 모델링 방법에서 사용되고 있는 입력변수의 수(2 또는 3)로 제한하여 제안된 모델에 적용한다. 전체 시스템 입력 데이터 쌍 296개 중 처음 148개의 데이터는 학습 데이터로 이용하고 나머지 148개의 데이터는 테스트 데이터로 나누어 제안된 모델의 근사화와 일반화 능력을 평가한다. 평가를 위한 성능지수는 식 (36)를 이용한다.

표 5는 GAs의 구조와 파라미터의 동조 방법에 따라 생성된 GoNFN의 성능지수를 보여준다. 결과에서 보여지듯이 앞의 예제들과 마찬가지로 구조와 파라미터를 동시에 동조하는 방법이 효과적이며, 입력변수는 3입력 $u(t-2), y(t-2), y(t-1)$ 를 선택하였을 때 좋은 출력결과를 얻을 수 있었다. 또한 ㉠에 의한 변수 선택보다는 ㉢에 의한 변수 선택이 적절한 시스템 입력변수임을 다른 연구결과와의 비교로부터 알 수 있다. 즉, 2입력 모델의 경우 ㉢방법에 의해 일반적으로 가스로

공정을 모델링하기 위해 가장 많이 사용되는 입력변수 $u(t-3), y(t-1)$ 가 선택되었다[4-6,14,18,19]. 가스로 공정에 대한 GoNFN의 모델로 ㉢의 방법에 의해 생성된 구조를 선택하였다. 선택된 GoNFN의 구조는 그림 7과 같고, GAs에 의한 모델의 최적화 과정, BP에 의한 모델의 학습과정을 그림 8과 그림 9에 보였다.

표 5 가스로 공정에 대한 GoNFN의 성능지수

Table 5 Performance index of GoNFN for the gas furnace

Model	Structure		Optimization		EPI	EPI	
	Inputs	Nodes	Opt.	Trng.			
㉠	GAs	2+2+2	Min-Max	GAs	BP	0.0225	0.144
	Tuned [u(t-2), y(t-2), y(t-1)]	2+2+2	GAs	Tuned	BP	0.0264	0.137
㉢	GAs	2+2+2+2	GAs	GAs	BP	0.0339	0.116
㉢	3 ≥ GAs	2+2+2	Min-Max	GAs	BP	0.0226	0.143
	Tuned [u(t-2), y(t-2), y(t-1)]	2+2+2	GAs	Tuned	BP	0.0257	0.131
㉢	3 ≥ GAs	2+2+2	GAs	GAs	BP	0.0228	0.136
	3 ≥ [u(t-2), y(t-2), y(t-1)]	3+3+3	GAs	GAs	BP	0.0251	0.133
㉡	2 ≥ GAs	2+2	Min-Max	GAs	BP	0.0286	0.334
	Tuned [u(t-3), y(t-1)]	2+2	GAs	Tuned	BP	0.0317	0.310
㉢	2 ≥ GAs	2+2	GAs	GAs	BP	0.0337	0.306

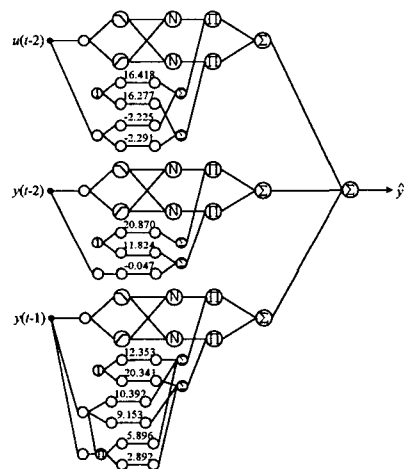


그림 7 가스로 공정에 대한 GoNFN 구조
Fig. 7 Topology of GoNFN for the gas furnace

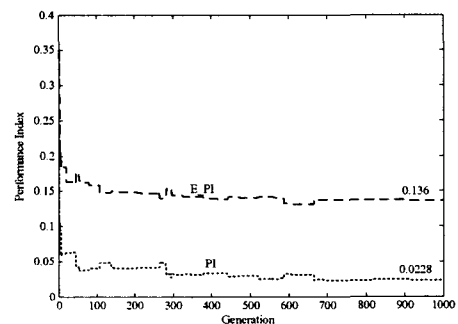


그림 8 GAs에 의한 GoNFN의 최적화 과정
Fig. 8 Optimal procedure of the GoNFN by GAs

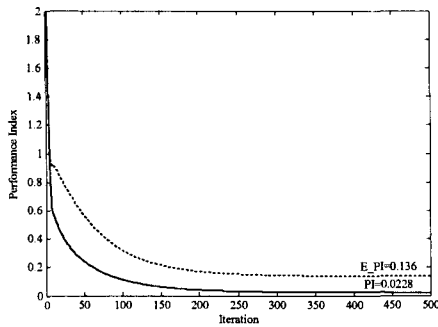


그림 9 GoNFN 구조의 학습과정
Fig. 9 Learning procedure of the GoNFN architectures

표 6 기존 모델방법과의 동정오차 비교
Table 6 Comparison of performance with other modeling methods

		Performance Index			Number of rules
Fuzzy	Complex[4]	Simplified	0.024	0.328	4(2×2)
		Linear	0.023	0.306	4(2×2)
	HCM[5]	Simplified	0.755	1.439	6(3×2)
		Linear	0.018	0.286	6(3×2)
	HCM+GAs[5]	Simplified	0.035	0.289	4(2×2)
			0.022	0.333	6(3×2)
		Linear	0.026	0.272	4(2×2)
			0.020	0.264	6(3×2)
	Hybrid[6] (GAs+Complex)	Simplified	0.024	0.329	4(2×2)
		Linear	0.017	0.289	4(2×2)
	Min-Max[23]	Simplified	0.022	0.335	4(2×2)
			0.022	0.336	6(3×2)
Linear		0.024	0.358	4(2×2)	
GAs[23]	Simplified	0.020	0.362	6(3×2)	
		0.023	0.344	4(2×2)	
	Linear	0.018	0.264	4(2×2)	
Neural Networks[5]		0.034	4.997		
Kim, et al.'s model[18]		0.034	0.244	2	
Lin and Cunningham's model[19]		0.071	0.261	4	
Oh's Adaptive FNN[25]		0.021	0.332	9(3×3)	
		0.022	0.353	4(2×2)	
FNN[14]	Simplified	0.043	0.264	6(3+3)	
	Linear	0.037	0.273	6(3+3)	
Multi-FNN[22]	Simplified	0.025	0.274	6(3+3)×4	
	Linear	0.024	0.283	6(3+3)×4	

표 6에서 기존의 모델과 제안된 모델들의 성능지수를 비교하였다. 퍼지모델은 입력 데이터의 최소, 최대값을 이용한 동정(Min-Max), GAs를 이용한 동정, Complex를 이용한 동정, 하이브리드 동정(GAs+Complex), HCM과 GAs를 이용한 동정 등 멤버쉽함수의 파라미터 동정과 관계된다. FNN은 하이브리드 알고리즘을 이용하여 멤버쉽함수의 파라미터를 동정한 구조이며, Multi-FNN은 클러스터를 이용한 구조로 4개의 FNN으로 구성된다. 즉, Multi-FNN의 구조를 위해 24개의 퍼지규칙이 사용된다. 결과의 비교로부터 제안된 구조는 기존의 모델들에 비해 데이터 특성에 맞는 다양한 구조를 설계함으로써 좋은 성능의 모델을 구축할 수 있었다.

5. 결론

본 논문은 고급 CI 기술의 한 영역으로써 효과적인 비선형 문제의 분석과 해결을 위한 진화론적 최적 뉴로퍼지 네트워크(Genetically optimized Neurofuzzy Networks; GoNFN)의 구조 개발과 광범위한 설계방법론을 토의하였다. 또한 실험예제를 통해 각 모델의 구조와 성능을 알아보았으며, 제안된 구조들의 비교와 기존의 모델링 기법들과의 결과 비교를 통해 설계방법론의 효율성과 모델구조의 특징을 알아보았다.

제안된 GoNFN 구조는 NFN의 최적화 구조로 GAs를 이용하여 구조 문제를 다루었다. NFN 구조는 입력변수의 퍼지관계(Fuzzy set; FS)를 기반으로 한 NFN 구조로 기존의 간략추론과 선형추론 형태의 네트워크 구조를 통합 및 확장하여 설계하였다. 실험예제를 통한 결과로부터 GoNFN은 구조와 파라미터를 동시에 고려한 방법이 효율적이고, 타당하며, 좋은 출력특성을 가지게 된다.

본 논문에서 제안된 GoNFN 구조는 GAs의 염색체 정보에 의해 시스템 환경의 변화에 적응하며, 진화함으로써, 유기적인 관계에 의해 유연성을 가진 구조로 발전하였고, 고급 CI 기술에 의해 다양한 구조로 설계되었다. 이들은 비선형 예제들을 통하여 주어진 문제에 대한 유연성과 정확성 및 객관성과 독창성을 입증하였고, 근사화와 예측(일반화) 능력이 우수한 특징을 가진다.

감사의 글

본 연구는 산업자원부의 지원에 의하여 기초전력연구원(R-2003-0-285) 주관으로 수행된 과제임.

참고 문헌

- [1] W. Pedrycz and J.F. Peters, Computational Intelligence and Software Engineering, World Scientific, Singapore, 1998.
- [2] L. W. Chan and F. Fallside, "An Adaptive Training Algorithm for Back Propagation Networks", Computer Speech and Language, Vol. 2, pp.205-218, 1987.
- [3] C. M. Bishop, Neural Networks for Pattern Recognition, Oxford Univ. Press, 1995.
- [4] 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.
- [5] B. J. Park, W. Pedrycz and S. K. Oh, "Identification of Fuzzy Models with the Aid of Evolutionary Data Granulation", IEE Proceedings-Control theory and application, Vol. 148, Issue 5, pp. 406-418, 2001.
- [6] S. K. Oh, W. Pedrycz and B. J. Park, "Hybrid Identification of Fuzzy Rule-Based Models", Inter. Journal of Intelligent Systems, Vol. 17, Issue 1, pp.77-103, 2002.
- [7] David E. Goldberg, Genetic Algorithms in search, Op-

timization&Machine Learning, Addison-wesley, 1989.

[8] Z. Michalewicz, Genetic Algorithms + Data Structure = Evolution Programs, Springer- Verlag, 1992.

[9] W. Pedrycz and G. Vukovich, "Granular Neural Networks", Neurocomputing, Vol. 36, pp.205-224, 2001.

[10] J. S. R. Jang, C. T. Sung and E. Mizutani, Neuro-Fuzzy and Soft Computing, Prentice Hall, 1997.

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

[12] 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.

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

[14] S. K. Oh, W. Pedrycz and H. S. Park, "Hybrid Identification in Fuzzy-Neural Networks", Fuzzy Sets and Systems, Vol. 138, No. 2, pp.399-426, 2003.

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

[17] D. E. Box and G. M. Jenkins, Time Series Analysis, Forecasting and Control, California: Holden Day, 1976.

[18] E. Kim, H. Lee, M. Park and M. Park, "A Simply Identified Sugeno-type Fuzzy Model via Double Clustering", Information Sciences, Vol 110, pp.25-39. 1998.

[19] Y. Lin, G. A. Cunningham III, "A new Approach to Fuzzy-neural Modeling", IEEE Transaction on Fuzzy Systems, Vol. 3, No. 2, pp. 190-197, 1997.

[20] 오성권, 프로그래밍에 의한 컴퓨터지능(퍼지, 신경회로망 및 진화알고리즘을 중심으로), 내하출판사, 2002.

[21] 진강규, 유전알고리즘과 그 응용, 교우사, 2000.

[22] 박호성, 오성권, "HCM 클러스터링에 의한 다중 퍼지-뉴럴 네트워크 동정과 유전자 알고리즘을 이용한 이의 최적화", 한국 퍼지 및 지능 시스템 학회 논문지, 10권, 5호, pp.487-496, 2000.

[23] 박병준, 오성권, 안태천, 김현기, "유전자 알고리즘과 하중값을 이용한 퍼지시스템의 최적화", 대한전기학회 논문지, 48A권, 6호, pp. 89-799, 1999.

[24] 안태천, 오성권, "발전소의 대기오염물질 배출패턴 모델 정립", 기초전력공학 공동연구소, 1997.

[25] 오성권, 박춘성, 박병준, "적응 퍼지-뉴럴네트워크를 이용한 비선형 공정의 온-라인 모델링", 대한전기학회 논문지, 48A권, 10호, pp.1293-1302, 1999.

저 자 소 개



박 병 준(朴 炳 俊)

1998년 원광대 제어계측공학과 졸업. 동대학원 석사(2000). 동대학원 박사(2003). 관심분야는 컴퓨터 지능(CI), 뉴로퍼지컴퓨팅, 퍼지이론, 하이브리드 시스템, 유전알고리즘, 지능형 모델링 및 제어, 시스템 자동화 등.

Tel : (063) 850-6342

E-mail : lcap@wonkwang.ac.kr



김 현 기(金 賢 起)

1977년 연세대 전기공학과 졸업. 동대학원 석사(1985), 동대학원 박사(1991).1987년~현재 수원대 전기공학과 교수, 2002년~현재 한국 산학연 기술학회 부회장 및 한국산학연 논문집 편집위원 . 관심분야는 뉴로퍼지컴퓨팅, 퍼지이론, 유전알고리즘, 지능형 모델링 및 제어, 자동화시스템 등.

Tel : (031) 220-2666

E-mail : hkkim@suwon.ac.kr



오 성 권(吳 聖 權)

1981년 연세대학교 전기공학과 졸업. 1983-89년 금성산전연구소(선임연구원). 1993년 연세대 대학원 전기공학과 졸업(공학박). 1996-1997년 캐나다 Manitoba대학 전기 및 컴퓨터공학과 Post-Doc. 1993년~현재 원광대 전기전자 및 정보공학부 부교수. 2002년~현재 대한전기학회 및 제어·자동화·시스템공학회 편집위원, 관심분야는 컴퓨터 지능(CI), 뉴로퍼지컴퓨팅, 퍼지이론, 하이브리드 시스템, 유전알고리즘, 지능형 모델링 및 제어, 시스템 자동화 등.

Tel : (063) 850-6342

E-mail : ohsk@wonkwang.ac.kr