

Short-Term Electrical Load Forecasting using Neuro-Fuzzy Models

朴永鎭* · 沈炫正** · 王普賢***
 (Young-Jin Park · Hyun-Jeong Shim · Bo-Hyeun Wang)

Abstract - This paper proposes a systematic method to develop short-term electrical load forecasting systems using neuro-fuzzy models. The primary goal of the proposed method is to improve the performance of the prediction model in terms of accuracy and reliability. For this, the proposed method explores the advantages of the structure learning of the neuro-fuzzy model. The proposed load forecasting system first builds an initial structure off-line for each hour of four day types and then stores the resultant initial structures in the initial structure bank. Whenever a prediction needs to be made, the proposed system initializes the neuro-fuzzy model with the appropriate initial structure stored and trains the initialized model. In order to demonstrate the viability of the proposed method, we develop an one hour ahead load forecasting system by using the real load data collected during 1993 and 1994 at KEPCO. Simulation results reveal that the prediction system developed in this paper can achieve a remarkable improvement on both accuracy and reliability compared with the prediction systems based on multilayer perceptrons, radial basis function networks, and neuro-fuzzy models without the structure learning.

Key Words : 전력 수요 예측, 뉴로-퍼지 모델, 구조 학습, 초기 구조 बैं크

1. 서 론

전력 계통의 운용 계획을 최적화 하기 위해서 연간 최대 전력수요와 시간별 전력수요에 대한 장단기간의 수요 예측에 관한 연구가 활발하게 진행중이다. 특히, 단기 수요 예측은 발전비용과 신뢰도에 크게 영향을 주며, 전력계통의 제어 및 단기계획, 경제급전, 전력조류계산 등의 입력자료로 활용되기 때문에 정확한 예측시스템의 개발이 절실하게 요구된다 [1]. 그러나 지수평활화법 또는 상태공간재구성기법과 같은 단기 수요 예측을 위한 전통적인 방법으로는 날씨와 같이 부하특성에 영향을 미치는 여러 가지 요인을 동시에 고려하고자 할 때 모델링 과정이 매우 복잡해지고, 주어진 문제의 비선형성이 증가하는 경우 예측의 정확도가 떨어지는 단점이 있다.

최근 이와 같은 문제를 극복하고자 신경회로망, 특히 다층 퍼셉트론 (Multilayer Perceptron, MLP)을 전력 수요 예측에 활용하는 연구가 집중적으로 수행되고 있다 [2-5]. 이들 연구는 MLP가 어떠한 비선형 함수도 근사화 할 수 있고, 유연한 입력 선택이 가능하며, 주어진 데이터로부터 학습에 의해 예측모델을 비교적 용이하게 구할 수 있다는 장

점을 기반으로 한다. 1992년에 Park은 MLP를 이용하여 단기 전력 수요 예측시스템을 구현할 수 있다는 가능성을 입증하였고 [2], 이후 온도 등과 같은 날씨 요인을 입력 변수로 활용하여 예측시스템의 성능을 향상하는 시도가 있었다 [3]. 최근에는 매우 어려운 문제로 인식되고 있는 특수일의 부하 예측을 위해서 다중 모듈 구조를 갖는 예측시스템에 관한 연구가 시도되고 있다 [4, 5].

특히, [5]에서 제시한 다중 모듈 구조를 갖는 MLP 기반 전력 수요 예측시스템은 지금까지 미국내 약 30개의 전력회사에서 실제 활용되는 성과를 거두고 있다. 그러나, 이와 같은 성공에도 불구하고 MLP를 기반으로 개발된 비선형 예측시스템은 기본적으로 다음과 같은 단점을 갖는다:

- MLP가 black box이기 때문에, 학습을 통해서 데이터로부터 유추된 예측모델의 규칙을 표현할 수 없다. 신경회로망으로부터 얻어진 예측결과를 해석하는 것이 불가능하게 되고, 결국 예측시스템의 실제 사용자는 예측결과의 신뢰성에 관해 지속적으로 의심 할 수밖에 없다.
- 최적의 성능을 얻기 위한 MLP의 은닉층의 수와 각 은닉층의 노드 수를 결정하는 효과적이며 체계적인 방법이 존재하지 않는다. 구조결정을 위한 체계적인 방법이 없기 때문에 MLP의 학습은 필연적으로 시행착오의 과정을 반복하여야 한다.

신경회로망의 학습능력과 퍼지시스템의 구조적 지식표현 능력을 융합하는 뉴로-퍼지 모델은 MLP의 단점을 극복하기 위한 효과적 대안으로 인식되고 있다. 1995년에 Bakirtzis는 뉴로-퍼지 모델의 장점을 단기 전력 수요 예측에 활용하기 위해 요일별, 시간별로 학습된 168개의 뉴로-

* 正 會 員 : 江陵大 電子工學科 碩士
 ** 準 會 員 : 江陵大 電子工學科 碩士課程
 *** 正 會 員 : 江陵大 電子工學科 助敎授 · 工博
 接受日字 : 1999年 10月 26日
 最終完了 : 2000年 2月 23日

퍼지 모듈을 사용하여 익일의 부하를 예측하였다 [6]. 이후 Mori는 36개의 퍼지 규칙을 갖는 뉴로-퍼지 모델을 이용하여 한 시간 앞의 전력수요를 예측하는 연구를 수행하였다 [7]. 그러나, 이들 연구는 예측을 위해 필요한 규칙의 수와 학습을 위한 초기 인자 값을 결정하는 구조학습에 관해 전혀 고려하지 않았다.

효과적인 뉴로-퍼지 모델을 구성하기 위해서는 체계적인 구조학습이 필수적이다. 뉴로-퍼지 모델의 구조학습은 관련된 입력 변수를 선정하는 문제와 입력 공간을 분할하는 문제로 구성된다. 입력 공간 분할 문제는 학습 데이터를 이용하여 가능한 적은 수의 규칙으로 요구되는 정확도를 확보할 수 있도록 뉴로-퍼지 모델의 초기 인자 값을 구하는 것이다. Sugeno는 뉴로-퍼지 모델의 효율성을 향상하기 위해서 뉴로-퍼지 모델링에서 구조학습에 관한 연구를 처음 시도하였다 [8]. 이후 Wang은 진화 연산을 이용하여 뉴로-퍼지 모델에서 입력 공간을 분할하는 체계적인 방법을 제안하였다 [9]. 한편 Kubat는 레이디얼 베이스 함수 네트워크 (Radial Basis Function Network, RBFN)을 분류 문제에 적용하는 경우에 대해 결정트리를 이용하여 입력 공간을 분할하는 방법을 제안하였다 [10].

본 논문은 뉴로-퍼지 모델의 구조학습을 이용하여 한 시간 앞의 전력 수요를 예측하는 단기 수요 예측시스템을 개발하기 위한 체계적인 방법을 제안한다. 제안된 방법은 일반적으로 부하의 특성이 평일(화,수,목,금), 월요일, 토요일, 일요일로 분류된다는 사실에 착안하여, 네 개의 요일 유형 각각에 대해 매 시간 별로 초기 구조를 구성한다. 초기 구조를 구성하기 위해서 Kubat이 제안한 결정 트리 방법을 이용하고 [10], 구성된 시간별 초기 구조는 초기 구조뱅크에 저장한다. 구조뱅크에 저장된 초기 구조 중에서 예측 시점에 필요한 초기 구조를 뉴로-퍼지 모델에 로드하여 모델을 초기화한다. 실제 상황에서 제안된 방법의 실효성을 검증하기 위해서 한국전력의 1993년과 1994년 실제 부하 데이터를 이용하여 모의 실험을 수행한다. 특히, 구조학습 없이 뉴로-퍼지 모델을 이용하여 구현한 예측시스템의 예측결과와 비교하여 구조학습이 적은 수의 퍼지 규칙으로도 향상된 성능을 얻는데 기여한다는 사실을 입증한다.

본 논문의 구성은 다음과 같다. 2장에서는 뉴로-퍼지 모델과 구조학습에 관해서 소개하고, 3장에서는 제안된 뉴로-퍼지 기반 전력 수요 예측시스템을 설명한다. 4장에서는 제안된 방법의 타당성을 검증하기 위해 모의 실험을 수행하고 실험 결과를 분석한다. 마지막으로 5장에서 결론을 제시한다.

2. 뉴로-퍼지 모델과 구조 학습

퍼지 규칙 베이스, 추론기, 비퍼지화기로 구성된 퍼지 추론시스템은 일련의 퍼지 규칙으로부터 퍼지 추론을 수행한다. 규칙 베이스를 구성하는 퍼지 규칙은 다음의 형태를 갖는다:

$$\text{규칙 1: } (x_1 \text{ is } A^1_1) \text{ and } (x_2 \text{ is } A^1_2) \text{ and } \dots \\ (x_n \text{ is } A^1_n), \text{ then } y \text{ is } q^1.$$

$$\text{규칙 2: } (x_1 \text{ is } A^2_1) \text{ and } (x_2 \text{ is } A^2_2) \text{ and } \dots \\ (x_n \text{ is } A^2_n), \text{ then } y \text{ is } q^2. \\ \dots \quad (1)$$

$$\text{규칙 } p: (x_1 \text{ is } A^p_1) \text{ and } (x_2 \text{ is } A^p_2) \text{ and } \dots \\ (x_n \text{ is } A^p_n), \text{ then } y \text{ is } q^p.$$

여기서 $x_j(1 \leq j \leq n)$ 는 입력 변수이고, y 는 출력 변수이다. A^i_j 와 $q^i(1 \leq i \leq p)$ 는 소속 함수에 의해 정의되는 퍼지 집합이며, 본 논문에서는 식 (1)에서 후건부 퍼지 집합이 상수의 형태를 갖는 간단한 구조의 퍼지 규칙을 고려한다.

2.1 뉴로-퍼지 시스템

뉴로-퍼지 시스템은 신경회로망의 구조를 이용하여 퍼지 추론시스템을 구현한다. Lin과 Lee는 5개의 층을 갖는 신경회로망을 이용하여 퍼지 추론시스템을 구현하여, 신경회로망의 학습능력과 퍼지시스템의 구조적 지식 표현 능력을 융합하는 시도를 하였다 [11]. Jang은 Sugeno 타입의 퍼지 규칙을 구현할 수 있는 ANFIS를 제안하여 뉴로-퍼지 시스템 연구를 체계화하였다 [12, 13]. 이후 여러 연구그룹에서 뉴로-퍼지 시스템에 관한 다양한 연구 결과를 발표하고 있다 [14-17].

그림 1은 두 개의 입력 변수와 하나의 출력 변수를 갖는 뉴로-퍼지 시스템의 구조이다. 두 개의 퍼지 규칙을 표현한 그림 1의 구조에서 입력 층의 입력 노드는 퍼지 규칙의 전건부 소속 함수 A^i_j 를 표현하며, 출력 층의 출력 노드는 후건부 소속 함수 q^i 를 표현한다. 입력 층의 입력 노드는 x 를 입력으로 받아 입력과 소속 함수와의 적합도를 계산한다. 만약 소속 함수로 Gaussian 함수를 사용하면, 적합도는 다음 식으로 계산된다:

$$\mu_{A^i_j}(x_j) = \exp \left[- \left(\frac{x_j - c^i_j}{\sigma^i_j} \right)^2 \right] \quad (2)$$

규칙 층의 규칙 노드는 계산된 적합도를 입력으로 받아 각 규칙의 발화 강도를 계산한다. 예를 들어, i 번째 규칙 노드의 출력은 다음 식에 의해 구해진다:

$$R_i(x) = \prod_{j=1}^n \mu_{A^i_j}(x_j) \quad (3)$$

뉴로-퍼지 시스템의 마지막 층은 비퍼지화 작업을 수행한다. 만약 퍼지 추론에서 무게중심법에 기반한 비퍼지화를 수행한다고 가정하면, 뉴로-퍼지 시스템의 최종 출력은 다음 식과 같다:

$$y = \frac{\sum_{i=1}^p R_i q^i}{\sum_{i=1}^p R_i} \quad (4)$$

식 (2)-식 (4)와 같이 주어진 뉴로-퍼지 시스템에서 출력을 구하는 과정은 주어진 퍼지 규칙으로부터 퍼지 추론을 수행

하는 과정과 동일하게 된다. 뉴로-퍼지 시스템의 인자 학습은 주어진 학습 데이터로부터 입력 노드의 전건부 소속 함수 A_i^j 와 출력 노드의 후건부 인자 q^j 를 설정된 목적 함수를 최소화하도록 정하는 것이다. 본 논문에서는 이를 위해 오차 역전파 학습을 이용한다 [18].

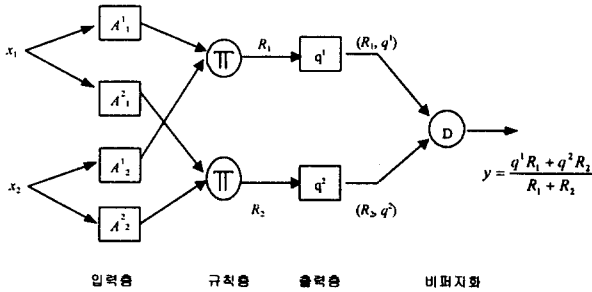


그림 1 뉴로-퍼지 시스템의 구조
Fig. 1 Architecture of a neuro-fuzzy system

2.2 뉴로-퍼지 모델의 구조학습

일반적인 모델링 문제에서와 같이 뉴로-퍼지 모델링에서도 구조학습은 모델의 성능을 향상하기 위해 필수적인 과정이다. 뉴로-퍼지 모델의 구조학습은 입력 변수를 선정하는 문제와 입력 공간을 분할하는 문제로 구성된다. 효율적인 모델을 구성하기 위해서 최적의 입력 변수를 선정하는 문제가 매우 중요하지만, 본 논문에서는 입력 공간 분할 문제를 주로 다루기로 한다. 지역 표현을 기반으로 하는 뉴로-퍼지 모델에서 입력 공간 분할 문제는 모델에서 사용하는 퍼지 규칙의 수와 초기 인자 값을 결정하는 문제로 귀결된다.

뉴로-퍼지 시스템에 저장된 일련의 퍼지 규칙은 입력 공간을 고차직사각형 (hyper-rectangle)으로 분할한다. 예를 들어, 두 개의 입력과 하나의 출력을 갖는 뉴로-퍼지 시스템이 하나의 퍼지 규칙을 학습했다고 가정하자. 학습된 퍼지 규칙은 2차원 입력 공간에서 그림 2와 같이 표현된다. 그림 2에서 $A_{1\alpha}$ 와 $A_{2\alpha}$ 는 학습된 퍼지 규칙의 전건부 소속 함수 A_1 과 A_2 의 α -cut 집합이다. 퍼지 규칙의 전건부를 표현하는 직사각형 분할은 α -cut 집합의 베이스를 원통형으로 확장한 $c(A_{1\alpha})$ 와 $c(A_{2\alpha})$ 의 support로부터 얻어진다:

$$H(\alpha) = S(c(A_{1\alpha})) \cap S(c(A_{2\alpha})) \quad (5)$$

여기서 직사각형 H 의 넓이는 α 의 함수이고, α 는 0보다 크고 1보다 작은 실수이다. 이러한 관점에서 n 개의 퍼지 규칙을 저장하는 구조의 뉴로-퍼지 시스템은 n 개의 고차직사각형을 이용하여 입력 공간을 분할한다고 해석할 수 있다.

만약 초기 입력 공간 분할에 대한 심각한 고려 없이 뉴로-퍼지 시스템을 학습하는 경우 우리는 일반적으로 필요한 퍼지 규칙의 수를 적당히 정하고, 인자 값을 랜덤하게 초기화하여 학습을 수행하게 된다. 이 경우 학습된 모델의 성능은 초기 인자 값에 영향받게 되고, 학습된 퍼지 규칙은 서로 중복되거나 모순되어 뉴로-퍼지 모델의 구조적 지식 표현이라는 장점을 상실하게 된다 [9].

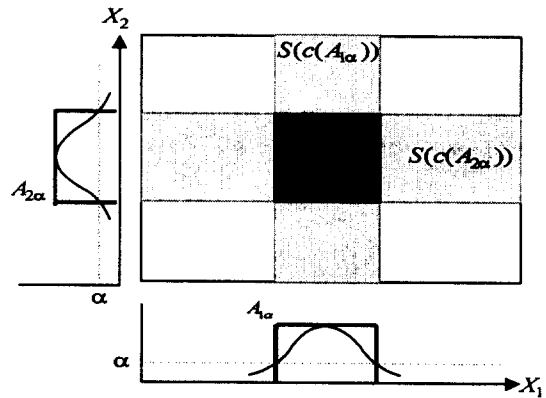


그림 2 입력 공간에 표현된 퍼지 규칙
Fig. 2 A fuzzy rule as a hyper-rectangle in the input space

이와 같은 문제를 해결하기 위해서 1993년에 Sugeno는 뉴로-퍼지 모델링에서 구조학습에 관한 연구를 처음 시도하였다 [8]. 이후 Sugeno 방법의 문제를 해결하고자 진화 연산을 이용한 입력 공간 분할 방법이 Wang에 의해 제안되었다 [9]. 한편 Kubat는 RBFN을 분류 문제에 적용하는 경우에 결정트리를 이용하여 입력 공간을 분할하는 방법을 제안하였다 [10]. 본 논문에서는 결정트리를 구성하는 작업이 매우 체계적이고 빠르다는 장점을 활용하기 위해 Kubat의 방법을 단기 수요 예측 문제에 적용하고자 한다. 이러한 이유에서 Kubat이 제안한 결정트리를 이용하여 뉴로-퍼지 모델의 구조를 학습하는 방법을 본 절에서 간단히 소개한다.

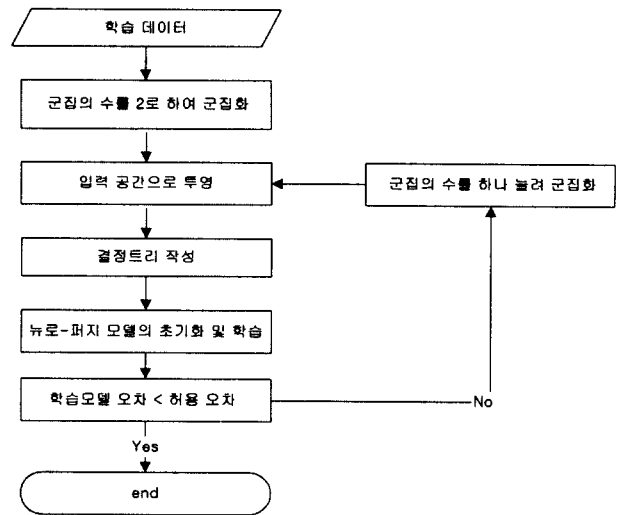


그림 3 결정트리를 이용한 구조학습의 구성도
Fig. 3 Procedure of structure learning using decision tree

그림 3은 Kubat이 제안한 결정트리에 기반한 입력 공간 분할 방법을 함수 근사화 문제에 적용하기 위해 수정한 전체 구성도이다. 수정된 방법은 기본적으로 출력 데이터를 군집화하고, 구해진 출력 군집을 입력 공간으로 투영하고, 마지막으로 결정트리를 작성한다. 제안된 입력 공간 분할의 목적이 적은 수의 규칙으로 향상된 정확도를 얻는 것이므로,

출력 데이터를 군집화할 때 군집의 수를 처음에는 2로 하고, 이후 원하는 성능을 얻을 때까지 군집의 수를 하나씩 늘려 가며 반복적으로 수행하게 된다.

2.2.1 출력 데이터의 군집화

Kubat이 제안한 뉴로-퍼지 모델의 입력 공간 분할 방법은 기본적으로 주어진 문제가 분류 문제라 가정한다. 본 논문에서는 함수 근사화 문제에 Kubat이 제안한 방법을 적용하기 위해서 학습 데이터의 출력부에 대해 군집화를 수행한다. 군집화를 위해서 k-means 알고리즘 또는 퍼지 c-means 방법을 사용할 수 있다. 어느 방법을 사용하는 군집화를 완료하면 각 출력 데이터가 속하는 군집이 구해진다:

$$y^i \in O_j \tag{6}$$

여기서 y^i 는 i 번째 학습 데이터의 출력이고, O_j 는 j 번째 출력 군집이다.

2.2.2 출력 군집을 입력 공간으로 투영

출력 군집을 입력 공간으로 투영하는 작업은 출력 군집 O_j 에 속한 모든 출력 데이터에 연관된 입력 데이터를 식별하는 것이다. 이 과정을 통해서 입력 데이터는 k 개의 군집으로 분할되고, 원래 주어진 함수 근사화 문제는 분류 문제로 변환된다.

2.2.3 결정트리 작성

출력 군집을 입력 공간으로 투영하여 얻은 입력 군집은 임의의 모양을 갖는다. 그러나, 입력 공간에서 고차직사각형 이외의 모양을 갖는 분할은 뉴로-퍼지 모델에 구현될 수 없기 때문에 결정트리를 이용하여 입력 군집의 모양을 고차직사각형으로 변환한다. 결정트리를 주어진 분류 문제에 적용하면 그림 4.(a)와 같은 이진 트리가 구성되고, 이는 입력 공간을 그림 4.(b)와 같은 트리 구조로 분할한 것으로 해석할 수 있다. 구성된 결정트리의 마지막 노드는 입력 공간상에서 하나의 퍼지 규칙이 되고, 마지막 노드의 수가 뉴로-퍼지 모델의 규칙 수가 된다.

2.2.4 뉴로-퍼지 모델의 초기화

구성된 결정트리를 이용하여 뉴로-퍼지 모델의 전건부 인자를 초기화 할 수 있다. 만약 Gaussian 함수를 이용하여 퍼지 규칙의 소속 함수를 표현하는 경우, Kubat이 제안한 아래의 방법으로 소속 함수의 중심과 넓이를 구한다 [10]. n 차원 입력공간을 분할한 n 차 직사각형은 각 차원에 하나의 소속 함수를 할당하여 표현된다 (그림 2 참조). 예를 들어, i 번째 고차 직사각형 H_i 의 j 번째 차원에 대해서 소속 함수의 중심 값 c_j^i 를 구한다고 가정하자.

- j 번째 차원의 한 끝점이 그림 5의 H_1 과 H_3 와 같이 x_{jmin} 이 나 x_{jmax} 이고, 다른 끝점은 다른 고차 직사각형의 끝점과 접해 있다면, 중심 값 c_j^i 는 다음과 같이 구해진다:

$$\begin{aligned} c_j^1 &= x_{jmin} \quad (H_1 \text{의 경우}) \text{ or} \\ c_j^3 &= x_{jmax} \quad (H_3 \text{의 경우}) \end{aligned} \tag{7}$$

- j 번째 차원의 양 끝점이 그림 5의 H_2 와 같이 다른 고차 직사각형의 끝점과 접해 있다면, 중심 값 c_j^2 는 아래와 같이 계산된다:

$$c_j^2 = \frac{x_{jmax}^2 + x_{jmin}^2}{2} \tag{8}$$

- j 번째 차원의 양 끝점이 그림 5의 H_4 와 같이 x_{jmin} 과 x_{jmax} 인 경우 소속 함수의 중심 값은 다음 식으로 구한다:

$$c_j^4 = \frac{x_{jmax} + x_{jmin}}{2} \tag{9}$$

H_i 의 j 번째 차원에 대한 소속 함수의 중심 값 c_j^i 에서부터 다른 어느 한 끝점까지의 거리를 I_j^i 라고 한다면 소속 함수의 넓이 σ_j^i 는 다음과 같이 계산된다:

$$\sigma_j^i = \frac{I_j^i}{[\ln(1/a)]^{1/2}} \tag{10}$$

여기서 a 는 식 (5)에서 고차직사각형을 정의할 때 사용한 α -cut 퍼지 집합의 인자이다.

마지막으로 뉴로-퍼지 모델의 후건부 인자는 다음의 방법으로 초기화한다. 예를 들어, H_i 에 포함된 데이터 중에서 m 번째 클래스에 속한 데이터 수가 최대라 하자. 이 경우 H_i 에 연관된 후건부 인자 q^i 는 다음과 같이 구해진다:

$$q^i = v_m \tag{11}$$

여기서, v_m 은 m 번째 출력 군집의 중심 값이다.

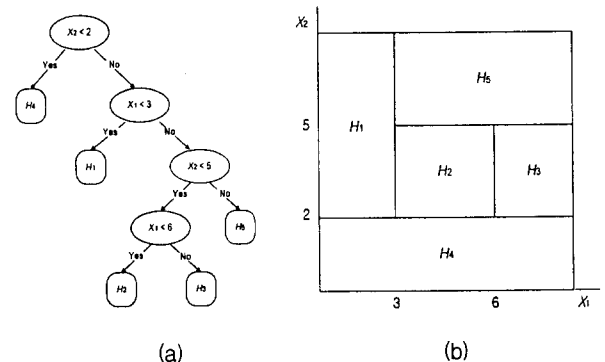


그림 4 결정트리와 입력 공간 분할. (a) 이진 트리의 예. (b) (a)의 이진 트리가 표현한 입력 공간의 분할

Fig. 4 Decision tree and input space partitioning. (a) An example of binary decision tree. (b) Partition of the input space represented by (a)

일단 뉴로 퍼지 모델의 초기화가 완료되면, 주어진 학습

데이터를 이용하여 초기화된 뉴로-퍼지 모델의 인자 학습을 수행한다. 만약 학습된 뉴로-퍼지 모델의 성능이 주어진 요구 조건을 만족하지 못하면, 그림 3에 표시한 것과 같이 군집의 수를 하나 증가시켜 2.2.1-2.2.4의 과정을 반복한다.

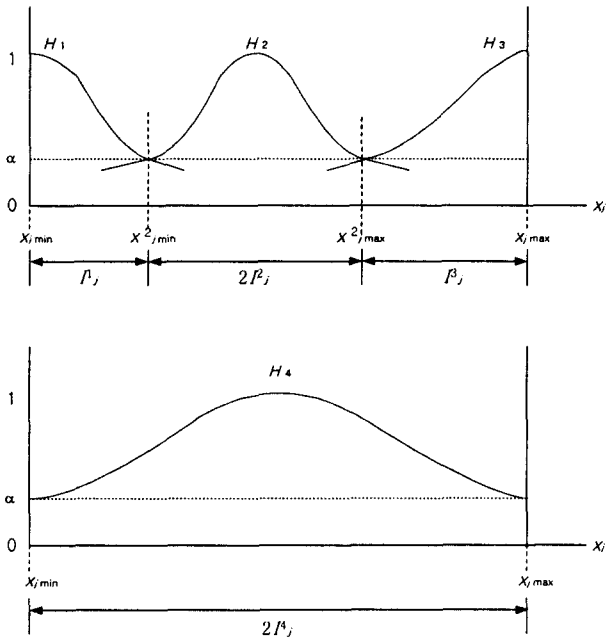


그림 5 입력 공간 분할 정보를 이용한 전건부 소속 함수의 초기화

Fig. 5 Initialization of membership functions using partitions in the input space

2.3 예제: 퍼지시스템의 재구성 문제

본 절에서는 퍼지시스템의 재구성 문제를 이용하여 결정트리에 기반한 뉴로-퍼지 모델의 초기화 방법을 설명한다. 또한 제안된 방법의 효과를 입증하고 구조학습의 중요성을 강조하기 위해 랜덤하게 초기값을 설정하고 학습한 뉴로-퍼지 모델과 성능을 비교한다

퍼지시스템 재구성은 미지의 퍼지시스템을 모델링하는 문제이다. 예를 들어, 그림 6과 같이 주어진 9개의 퍼지 규칙으로 구성된 퍼지시스템을 고려하자. 퍼지 규칙은 식 (1)과 같은 형태를 갖고, 전건부 퍼지 집합은 Gaussian 소속 함수에 의해 표현된다.

학습 데이터는 [-1,1]×[-1,1]의 범위에서 일정한 분포를 갖는 랜덤 입력 200개를 이용하여 수집한다. 입력 공간 분할 문제를 해결하기 위해 먼저 수집된 200개의 학습 데이터를 이용하여 출력 데이터를 k-means 알고리즘을 사용하여 군집화한다. 예를 들어, 군집의 수를 3으로 하여 군집화를 수행하고, 이를 입력 공간에 투영한 결과는 그림 7과 같다. 입력 군집은 임의의 모양을 갖고 있으며, 뉴로-퍼지 모델을 초기화하기 위해서 결정트리를 이용하여 군집의 모양을 고차원서각형으로 변환해야 한다는 것을 알 수 있다. 본 논문에서는 Quilan이 제안한 c4.5를 이용하여 결정트리를 구성한다 [19]. c4.5의 pruning 기능을 이용하여 퍼지 규칙의 수를

3에서부터 7까지 조정하고, 각 경우에 대해 뉴로-퍼지 모델을 초기화하여 오차 역전파 방법으로 학습한다. 학습율은 0.005로 고정하였고, 1000 epoch에서 학습을 끝냈다.

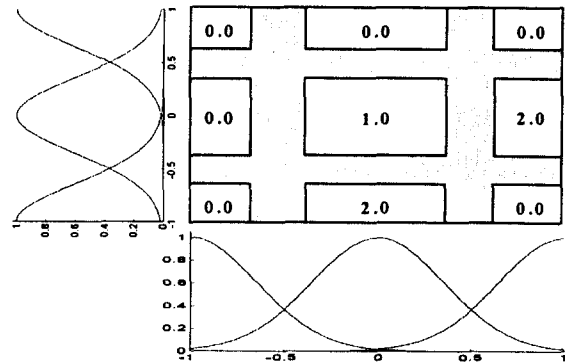


그림 6 재구성 문제에서 사용한 퍼지 규칙 베이스
Fig. 6 Fuzzy rulebase for fuzzy system reproduction

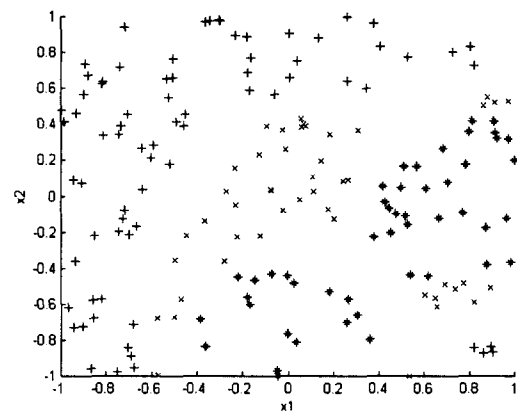


그림 7 출력 군집을 입력 공간으로 투영한 결과 (+: class 1, x: class 2, *:class 3)

Fig. 7 Projection of output clusters into the input space (+: class 1, x: class 2, *:class 3)

제안된 입력 공간 분할 방법의 효과를 검증하기 위해서 뉴로-퍼지 모델의 인자를 랜덤하게 초기화하여 학습한다. 비교를 위해 규칙의 수를 3에서 7까지 증가시켰고, 각 경우 50번 반복하여 모델을 학습한다. 표 1은 결정트리를 이용하여 뉴로-퍼지 모델을 초기화하고 학습한 경우와 랜덤하게 초기화하여 학습한 경우에 대한 뉴로-퍼지 모델의 정확도를 나타낸다. 랜덤하게 초기화한 뉴로-퍼지 모델에서 성능은 각 규칙 수에 대하여 50번 반복 실험하여 얻은 50개 모델의 정확도를 평균한 값이다. 모델의 정확도는 MAE (Mean Absolute Error)로 측정한다:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \tag{12}$$

표 1에서 알 수 있듯이 규칙의 수가 증가하면서 결정트리를

이용하여 구조학습을 수행한 뉴로-퍼지 모델이 구조학습을 전혀 고려하지 않은 모델보다 더 우수한 성능을 보인다.

그림 8은 6개의 규칙을 갖는 뉴로-퍼지 모델에 학습된 퍼지 규칙을 입력 공간에 표현한다. 구조학습을 수행한 경우는 그림 6에서 주어진 퍼지시스템의 규칙을 거의 완벽하게 재구성하고 있으나, 랜덤하게 초기화하여 학습한 뉴로-퍼지 모델은 그렇지 못함을 알 수 있다. 결론적으로 제안된 입력 공간 분할 방법을 이용하면 적은 수의 규칙으로 요구되는 정확도를 만족하는 모델의 구성이 가능할 뿐만 아니라, 구조적 지식 표현이라는 뉴로-퍼지 모델의 장점을 유지할 수 있음을 알 수 있다. 다음 장에서는 제안된 뉴로-퍼지 모델의 입력 공간 분할 방법을 적용하여 단기 전력 수요 예측 시스템을 구현하는 방법에 대해 설명한다.

표 1 재구성 문제에서 초기화 방법에 따른 뉴로-퍼지 모델의 성능 비교 (ρ 는 규칙의 수)

Table 1 Performance comparison of neuro-fuzzy models (ρ is the number of fuzzy rules)

모델 \ ρ	3	4	5	6	7
랜덤하게 초기화한 모델	0.1429	0.1074	0.0753	0.0670	0.0465
결정트리 기반 모델	0.1611	0.0837	0.0439	0.0107	0.0106

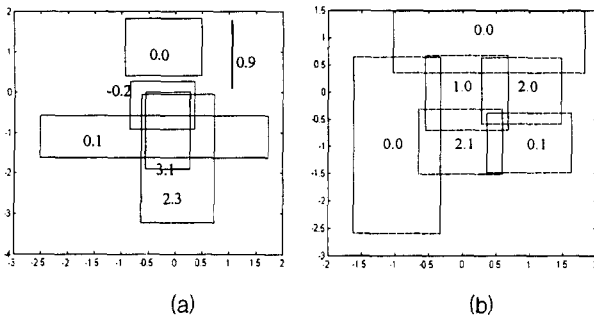


그림 8 학습된 뉴로-퍼지 모델의 퍼지 규칙. (a) 랜덤하게 초기화하여 학습한 경우. (b) 제안된 방법으로 초기화 한 경우

Fig. 8 Fuzzy rules encoded in neuro-fuzzy models. (a) Random initialization. (b) Initialization by the proposed method

3. 뉴로-퍼지 모델을 이용한 단기 전력 수요 예측

본 논문에서는 뉴로-퍼지 시스템을 실시간 학습하여 1시간 앞의 전력 수요를 예측하기 위한 단기 전력 수요 예측 시스템을 제안한다. 특히, 2장에서 논의한 뉴로-퍼지 시스템의 구조학습을 전력 수요 예측시스템에 적용하기 위해서 초기 구조 बैं크를 이용하는 방법을 제안한다. 초기 구조 बैं크는 평일, 토요일, 일요일, 월요일로 구분된 각 요일 유형에 대해 매 시간별로 구조 학습을 수행하여 생성한 96개의 초기 구조로 구성된다. 뉴로-퍼지 모델을 학습하여 예측시스템을 구축하기 위해서 그림 9와 같이 초기 구조 बैं크에 미리 저장된 초기 구조 중에서 예측 시점에 해당하는 초기 구조를

로드하여 뉴로-퍼지 모델을 실시간 초기화하고 학습하고 예측을 수행한다. 예를 들어, 예측시스템이 화요일 01시의 전력 수요를 예측하는 경우를 고려하자. 이를 위해 먼저 구조 बैं크로부터 평일 01시의 초기 구조를 선택하여 뉴로-퍼지 모델을 초기화한다. 데이터 베이스로부터 01시 수요 예측에 필요한 학습 데이터를 이용하여 초기화된 뉴로-퍼지 모델을 학습한다. 마지막으로 예측에 필요한 입력을 학습된 모델에 가하여 전력 수요 예측을 수행하게 된다.

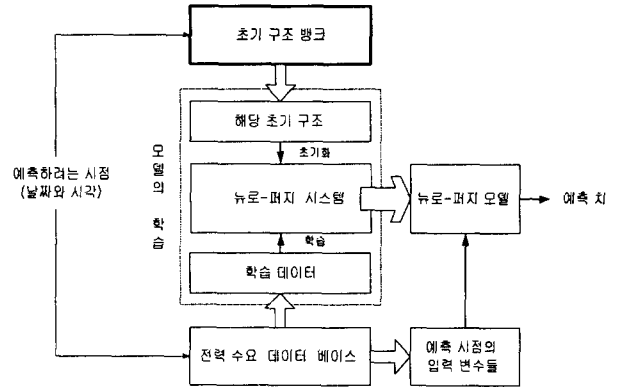


그림 9 뉴로-퍼지 모델을 이용한 전력 수요 예측 시스템의 구조

Fig. 9 Electrical load forecasting system using neuro-fuzzy model

표 2 입력 변수와 출력 변수

Table 2 Input variables and an output variable

입력 변수	$\rho(i, t-1), \rho(i, t-2)$ $\rho(i-1, t), \rho(i-1, t-1), \rho(i-1, t-2)$ $\rho(i-2, t), \rho(i-2, t-1), \rho(i-2, t-2)$ $\rho(i-7, t), \rho(i-7, t-1), \rho(i-7, t-2)$ $\rho(i-14, t), \rho(i-14, t-1), \rho(i-14, t-2)$
출력 변수	$\rho(i, t)$

예측시스템의 설계는 예측 모형의 입력 변수를 선정하는 작업에서 출발한다. 입력 변수를 선정하는 작업은 예측시스템의 성능 향상에 결정적인 역할을 한다. 그러나, 최적의 입력 변수를 선정하는 문제는 일반적으로 매우 어려운 문제로 간주된다 [8, 20]. 또한 본 논문의 목적이 효과적인 구조학습을 통해 예측시스템의 성능을 향상할 수 있다는 사실을 입증하는 것이므로 [4]에서 사용한 14개의 입력 변수를 그대로 사용한다. 표 2는 본 논문에서 사용한 입력 변수를 요약한다. i 는 예측하는 날을, t 는 예측하는 시점을 각각 의미한다. 그러므로, $\rho(i, t)$ 는 i 날 t 시각의 전력 수요가 된다.

일반적으로 하루 동안의 전력 수요 패턴은 화, 수, 목, 금의 평일, 월요일, 토요일, 일요일, 특수일의 유형으로 분류된다 [1]. 월요일 오전의 전력 수요는 일요일에 의해 영향을 받기 때문에 평일 (화, 수, 목, 금)의 전력 수요보다 작게 되고, 토요일 오전의 전력 수요는 평일의 전력 수요와 비슷하지만, 오후에는 평일 수요보다 대체로 낮게 나타난다. 일요

일의 부하는 다른 요일의 부하보다 훨씬 작고, 국경 공휴일이나, 설날, 추석 등과 같은 특수일의 경우는 다른 요일의 부하보다 작으면서 일요일과는 전혀 다른 수요 패턴을 갖는다.

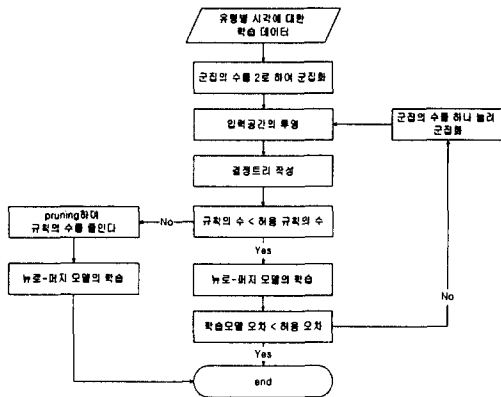


그림 10 초기 구조 생성 알고리즘
Fig. 10 Algorithm for creating initial structures

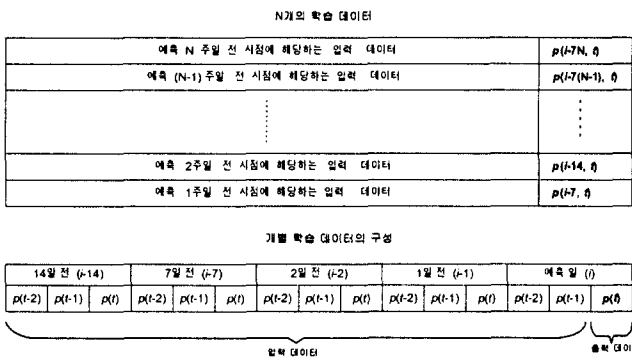


그림 11 학습 데이터의 생성 방법
Fig. 11 Method to generate training data

이와 같은 사실에 기초하여 본 연구에서는 특수일을 제외한 4개의 수요 패턴 유형에 대한 초기 구조 백크를 구축한다. 4개의 요일 유형에 대해서 시간별로 24개의 서로 다른 초기 구조를 갖기 때문에 초기 구조 백크는 모두 96개의 초기 구조로 구성된다. 하나의 초기 구조를 생성하기 위해서 필요한 학습 데이터는 해당 요일 유형 및 해당 시간에 대해 개별적으로 수집한다. 예를 들어, 평일 유형 01시에 대한 초기 구조를 만들기 위한 학습 데이터는 어느 특정한 한 주의 화, 수, 목, 금요일의 01시에 대한 학습 데이터를 모두 합하여 구성한다. 나머지 유형(토, 일, 월)의 초기 구조를 위한 학습 데이터는 2주의 데이터로부터 준비한다.

일단 초기 구조 생성을 위한 학습 데이터가 준비되면, 그림 10의 초기 구조 선정 알고리즘에 의하여 초기 구조 96개를 생성한다. 하나의 초기 구조를 생성하기 위해서 먼저 군집의 수를 2로 하고, 2.2절의 구조학습 방법으로 초기 구조를 만든다. 만약 생성된 규칙의 수가 허용 규칙의 수보다 적으면, 뉴로-퍼지 모델을 초기화하고 학습한다. 학습된 뉴

로-퍼지 모델의 오차가 허용 오차보다 작을 경우 작업을 종료하고, 그렇지 않은 경우 군집의 수를 하나 늘려 초기 구조를 다시 만든다. 만약 규칙의 수가 허용 규칙의 수보다 크면, 결정트리를 pruning하여 규칙의 수가 허용 규칙의 수보다 적은 초기 구조를 만들고 뉴로-퍼지 모델을 학습한 후 작업을 종료한다.

한 시간 앞의 부하를 예측하기 위해 뉴로-퍼지 모델을 매 시간 학습한다. 이를 위해 매 시간 해당되는 초기 구조를 초기 구조 백크에서 다운 로드 받아 뉴로-퍼지 시스템을 초기화한다. 초기화된 뉴로-퍼지 모델을 학습하기 위해서 표 2에서 제시한 14개의 입력변수를 기초로 그림 11과 같이 학습 데이터를 수집한다. 일반적으로 학습 데이터의 수는 임의로 정할 수 있다. 예를 들어, [4]에서는 단지 3개의 학습 데이터를 사용하였고, 본 논문 4장의 모의 실험에서는 50개의 학습 데이터를 사용하였다. 만약 학습 데이터를 준비할 때, 데이터가 특수일에 속하게 되는 경우에는 한 주일 전의 데이터로 대체한다. 이것은 모순되는 데이터의 학습을 피하기 위해서이다.

4. 모의 실험 결과 및 분석

제안하는 예측시스템의 성능을 평가하기 위하여 94년 1월에서 4월까지의 전력 수요를 예측하였다. 뉴로-퍼지 모델을 매 시간 학습하기 위해서 표 2에서 제시한 입출력 변수를 토대로 그림 11과 같이 50개의 학습 데이터를 구성했다. 50개의 학습 데이터를 준비하기 위해서는 보통 예측 시점부터 과거 약 1년 정도의 데이터가 필요하다.

초기 구조 백크를 구축하기 위해서 표 3에서 보여지는 것처럼 94년 1월 8일부터 1월 17일에 해당하는 학습 데이터를 사용했다. 평일 유형의 초기 구조를 학습하기 위해서 200개의 학습 데이터가 사용되었고, 나머지 유형의 초기 구조를 학습하기 위해서는 100개의 학습 데이터를 사용하였다. 구조학습을 위한 학습 데이터를 사용하여 그림 10에서 제시한 초기구조 백크 설계 방법에 따라 96개의 초기 구조를 결정한다. 표 4는 허용 규칙의 수를 10, 허용 오차를 1%로 하여 생성한 초기 구조의 규칙 수와 결정된 초기 구조를 이용하여 뉴로-퍼지 모델을 학습한 경우 얻어진 학습 데이터에 대한 오차를 나타낸다. 평일 유형에 대한 초기 구조는 평균 4.5개의 규칙을 갖고 토요일, 일요일, 월요일 유형에 대해서는 평균 4.2, 4.2, 4.0개의 규칙을 갖는다. 모델의 성능 평가를 위해서 평균 퍼센트 오차(Mean Absolute Percentage Error, MAPE)를 사용하였다:

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{\text{실제부하값} - \text{예측값}}{\text{실제부하값}} \right| \times 100 \quad (13)$$

예측을 수행하고자 하는 요일의 유형 및 시각에 해당하는 초기 구조를 선택하여 뉴로-퍼지 모델의 구조와 인자 값을 초기화한다. 준비된 50개의 학습 데이터를 이용하여 초기화된 뉴로-퍼지 모델을 학습하고, 구축된 예측시스템을 사용하여 예측을 수행하였다. 구축된 예측시스템의 성능을 1994년 1월 18일부터 1994년 4월 30일까지 테스트하여 월별, 요일별, 시간별로 표 5, 6, 7에 요약하였다. 본 논문에서 개발

한 예측시스템은 특수일에 대해 전혀 고려하지 않았기 때문에, 특수일에 대한 예측 결과를 제외하고 성능을 분석하였다.

표 3 초기 구조 बैं크를 위한 학습 데이터

Table 3 Training data set for the initial structure bank

유형	날짜	기간	학습 데이터 수
평일 (화, 수, 목, 금)	94년 1월 11일 ~94년 1월 14일	4일	200
토요일	94년 1월 8일 94년 1월 15일	2일	100
일요일	94년 1월 9일 94년 1월 16일	2일	100
월요일	94년 1월 10일 94년 1월 17일	2일	100

표 4 결정된 초기구조의 규칙 수와 학습 오차율

Table 4 Number of fuzzy rules and training error of the initial structures determined

유형	평일		토요일		일요일		월요일	
	규칙 수	학습 오차율	규칙 수	학습 오차율	규칙 수	학습 오차율	규칙 수	학습 오차율
1	8	1.2582	9	0.9897	7	1.3847	8	1.2378
2	6	0.8089	6	0.7645	4	0.7896	9	0.9639
3	4	0.5557	2	0.6395	3	0.8809	2	0.6884
4	5	0.6315	2	0.8007	4	0.7444	2	0.7166
5	5	0.6820	2	0.5954	2	0.8382	3	0.8900
6	4	0.7211	4	0.7487	4	0.9651	4	0.8921
7	5	0.8559	4	0.7261	5	0.8691	3	0.9783
8	3	0.8273	4	0.8178	7	0.7855	8	0.8748
9	6	0.9157	7	0.9653	9	1.0917	3	1.1178
10	7	0.1121	9	1.2079	4	0.9249	8	1.7299
11	2	0.7997	6	0.9108	4	0.9033	3	0.7286
12	4	0.9319	2	0.6221	2	0.6873	3	0.8684
13	4	0.6496	2	0.8143	2	0.9319	3	0.9864
14	4	0.8313	2	0.6589	2	0.9234	2	0.8523
15	4	0.8047	2	0.7157	2	0.7657	3	0.9360
16	4	0.6539	2	0.5706	2	0.6450	3	0.8166
17	4	0.6266	2	0.7396	2	0.9059	3	0.7557
18	5	1.2510	5	1.1429	7	1.0665	3	0.9330
19	4	0.8850	8	0.9734	7	0.9658	3	0.7172
20	3	0.8649	4	0.9954	4	0.7873	3	0.8237
21	7	0.8892	8	0.8825	4	0.7869	3	0.9274
22	3	0.7370	2	0.5825	4	0.6499	3	0.9078
23	4	0.7579	4	0.8775	4	0.7866	6	0.6816
24	4	0.8678	2	0.9472	6	0.9880	4	0.6868
평균	4.5	0.8300	4.2	0.8204	4.2	0.8778	4.0	0.9046

월별 예측 결과를 요약한 표 5에서 알 수 있는 것과 같이 예측의 성능이 일반적으로 한 시간 앞의 전력 수요를 예측할 때 요구되는 오차율인 1%정도 수준이었다. 또한, 3월과 4월의 예측 결과가 1월과 2월의 예측 결과보다 MAPE의 평균과 MAPE의 표준편차 측면에서 모두 우수하였다. 표 6의 요일별 예측 결과를 살펴보면, 월요일과 일요일이 다른 요일보다 예측 성능이 좋지 않게 나타났다. 시간별 예측 결과를 보면 출근시간(8시와 9시)과 퇴근시간(18시, 19시, 20시)에

비교적 예측 성능이 저하되었다. 또한, 시간별 예측 결과로부터 초기 구조의 학습 과정에서 얻어진 학습 성능(표 4 참조)이 예측 시스템의 실제 결과를 잘 반영하고 있음을 알 수 있다.

표 5 모의 실험 결과 (월별 분석)

Table 5 Simulation results (Month by month analysis)

제안된 방법 (Tr = 50)	월	1월	2월	3월	4월
	Ave	1.18	1.22	0.86	0.90
Std	0.94	1.48	0.83	0.96	
MLP (Tr = 3)	Ave	1.89	1.98	1.26	1.52
	Std	1.56	2.37	1.29	1.28
MLP (Tr = 50)	Ave	1.43	1.69	0.89	0.88
	Std	2.30	2.36	0.99	0.93
RBFN (Tr = 50)	Ave	1.57	1.92	1.01	0.99
	Std	2.18	2.88	1.10	1.10

Tr은 학습에 사용한 데이터의 수

표 6 모의 실험 결과 (요일별 분석)

Table 6 Simulation results (Day by day analysis)

제안된 방법 (Tr = 50)	요일	월	화	수	목	금	토	일
	Ave	1.19	0.83	0.81	0.86	0.83	0.92	1.55
Std	1.12	0.75	0.74	0.75	0.75	0.78	1.86	
Max	5.83	3.60	3.64	4.24	5.21	4.99	11.87	
MLP (Tr = 3)	Ave	1.61	1.29	1.10	1.24	1.30	1.21	3.03
	Std	1.45	1.04	0.90	0.98	1.49	0.93	2.54
	Max	6.87	4.98	4.10	5.13	7.52	4.16	14.23
MLP (Tr = 50)	Ave	1.48	1.20	0.90	0.85	0.94	0.90	1.73
	Std	2.61	1.73	0.81	0.82	1.14	0.85	2.16
	Max	36.07	17.41	6.05	7.39	13.85	6.05	12.73
RBFN (Tr = 50)	Ave	1.70	1.55	1.04	0.99	1.19	0.97	1.48
	Std	2.28	2.32	1.12	1.31	2.14	0.94	1.79
	Max	19.92	19.58	7.28	11.99	24.72	6.29	16.17

표 7 모의 실험 결과 (시간별 분석)

Table 7 Simulation results (Hour by hour analysis)

모델	제안된 방법 (Tr=50)			MLP (Tr = 3)			MLP (Tr = 50)			RBFN (Tr = 50)		
	Ave	Std	Max	Ave	Std	Max	Ave	Std	Max	Ave	Std	Max
1	1.14	1.02	5.21	1.33	1.24	6.64	1.81	4.04	36.07	3.31	3.92	24.72
2	0.76	0.59	3.08	1.32	1.28	6.87	1.07	0.99	7.42	1.15	1.49	10.49
3	0.64	0.58	3.20	1.34	1.35	6.75	0.86	0.77	4.76	0.73	0.69	3.37
4	0.60	0.61	3.52	1.30	1.33	6.03	0.92	0.96	5.09	0.75	0.75	3.95
5	0.66	0.63	3.30	1.37	1.27	6.43	0.87	0.81	4.31	1.05	1.23	5.74
6	0.75	0.71	4.69	1.22	1.23	6.85	0.97	0.92	5.59	1.13	1.40	8.19
7	1.12	0.92	5.83	1.32	1.16	6.53	1.10	0.92	5.48	1.49	2.36	19.92
8	1.39	1.19	7.05	1.49	1.32	7.60	1.24	0.97	4.05	1.96	2.48	17.57
9	1.87	1.39	8.84	1.45	1.50	10.83	1.34	1.56	10.33	2.06	2.22	14.76
10	0.98	1.37	11.87	1.58	1.88	12.95	1.52	2.39	17.41	2.21	2.67	16.33
11	0.84	1.31	11.28	1.71	2.10	14.23	1.22	1.91	12.04	1.09	2.06	16.17
12	0.73	1.03	9.33	1.75	2.16	13.49	1.00	1.63	11.45	0.70	0.65	4.72
13	0.88	0.81	6.61	1.81	1.89	11.13	0.92	0.93	6.72	1.06	1.37	8.52
14	0.80	0.89	7.61	1.84	1.96	11.93	0.97	1.23	7.24	0.79	0.87	6.74
15	0.64	0.93	8.58	1.88	1.89	12.16	0.91	1.21	6.73	0.81	0.97	6.27
16	0.75	0.93	8.04	1.82	1.80	11.08	0.83	1.08	8.08	0.74	0.87	6.75
17	0.76	0.96	8.58	1.91	1.73	10.37	0.99	1.83	11.48	0.93	0.92	5.27
18	1.58	1.25	5.79	2.05	1.62	7.70	2.00	2.22	11.55	2.04	2.35	19.58
19	1.07	0.88	4.83	2.10	1.57	6.54	1.11	1.01	5.10	1.38	1.31	9.55
20	2.25	1.34	7.67	1.44	1.22	7.44	1.18	1.09	5.73	1.14	1.12	7.17
21	0.71	0.72	4.43	1.32	1.20	7.44	1.11	1.55	12.73	0.98	1.02	6.89
22	0.92	0.92	5.35	1.18	1.25	7.51	0.92	0.91	6.57	0.67	0.46	1.97
23	0.92	0.98	5.93	1.11	1.13	6.82	1.14	1.18	5.59	1.07	0.92	4.95
24	1.11	1.00	4.94	1.23	1.13	5.85	1.10	1.30	8.82	0.96	0.76	3.56

4.1 제안된 방법과 MLP 예측 시스템의 성능 비교

제안한 예측시스템의 성능을 기존의 방법과 비교하여 검증하기 위해서 MLP를 이용한 예측시스템과 성능을 비교하였다. 공정한 비교를 위하여 MLP 기반 예측시스템의 입력 변수는 뉴로-퍼지 기반 예측시스템에서 사용한 14개의 입력 변수를 동일하게 사용하였다. MLP의 구조는 [4]에서 제시한 2개의 은닉 층을 갖는 14-5-3-1 구조를 사용하였다. [4]에서 언급한 부하 폭선의 베이스 라인 예측치중 한 시간 앞의 예측치를 구하기 위해 매 시간 3개의 학습 데이터를 사용하여 MLP를 학습하였다. 또한, 본 논문에서 제안한 뉴로-퍼지 기반 예측 시스템이 매 시간 50개의 학습 데이터를 이용하였기 때문에 공정한 비교를 위해서 MLP의 학습 데이터를 50개로 늘려 반복 실험하였다.

표 5에서 알 수 있듯이 제안된 방법이 매월 평균 MAPE에서 MLP 예측시스템보다 우수했다. 특히, 제안된 예측시스템의 MAPE의 표준 편차가 다른 두 개의 MLP 기반 예측시스템보다 우수함을 알 수 있었다. 이러한 결과를 표 6과 표 7에서 요약한 요일별, 시간별 분석 결과에서도 확인할 수 있었다. 이와 같이 향상된 성능을 얻게 되는 주된 이유는 제안된 구조 학습 방법이 학습된 신경회로망에서 overfitting을 줄일 수 있기 때문이라 판단된다. 결론적으로 보다 더 신뢰할 수 있는 예측시스템을 개발하기 위해 제안된 구조학습이 필수적이라는 사실을 본 비교 실험에서 확인할 수 있다.

4.2 제안된 방법과 RBFN 예측 시스템의 성능 비교

RBFN은 뉴로-퍼지 시스템과 마찬가지로 지역 표현을 한다. 특히, RBFN은 은닉 층의 인자를 변화시키지 않았을 때, 학습 속도가 빠르다는 장점이 있다. 이러한 장점을 활용하여 예측 시스템을 구현할 경우 어느 정도의 예측 성능이 나오는지 확인하기 위해서 RBFN을 이용하여 예측시스템을 구축하였다.

RBFN 기반 예측 시스템을 학습하기 위하여 매 시간 50개의 학습 데이터를 사용하였다. RBFN을 초기화하기 위해서 [18, 21]에서 제안한 방법을 사용했다. 이 방법은 기본적으로 학습 데이터 하나에 대해서 하나의 베이스스 함수를 할당하며, 베이스스 함수의 넓이는 p -heuristic 방법 ($p=40$)을 사용하여 구하였다.

표 5, 6, 7에서 알 수 있듯이 제안된 뉴로-퍼지 예측시스템이 RBFN 예측시스템보다 훨씬 더 적은 규칙을 사용했음에도 불구하고, MAPE의 평균과 MAPE의 표준 편차에서 우수함을 알 수 있다. 또한, RBFN 예측시스템의 성능을 MLP 예측 시스템과 비교하면, 학습 데이터 3개일 때보다는 우수하지만, 50개일 때보다는 떨어짐을 확인할 수 있다.

4.3 랜덤하게 초기화하여 학습한 뉴로-퍼지 시스템과의 성능 비교

마지막으로 뉴로-퍼지 모델의 구조 학습이 효과적이라는 사실을 입증하기 위해서 뉴로-퍼지 모델의 인자를 랜덤하게 초기화하여 뉴로-퍼지 예측시스템을 구축하였다. 비교를 위

해서 사용된 학습 데이터는 뉴로-퍼지 예측시스템과 동일하고, 규칙의 수는 표 4에서 제시한 것과 같이 한다. 랜덤하게 구축된 예측시스템의 성능을 표 8에 요약한다. 매우 좋지 않은 예측 결과를 얻었음을 알 수 있는데, 이는 뉴로-퍼지 모델을 이용하여 예측시스템을 설계할 때 사용하는 규칙의 수가 적을 경우 뉴로-퍼지 모델의 초기 인자 값을 랜덤하게 정하고 학습하는 방법이 현실성이 없음을 입증한다. 궁극적으로는 구조학습이 적은 수의 규칙으로도 모델의 정확도와 신뢰도를 확보할 수 있음을 반증한다.

표 8 랜덤 기반 뉴로-퍼지 예측 시스템의 월별 성능
Table 8 Prediction performance of the neuro-fuzzy model randomly initialized

모델	월	1월	2월	3월	4월
NFM (rand) Tr = 50	Ave	26.89	24.12	23.97	25.03
	Std	40.11	39.23	40.19	41.78

비교 실험을 통해서 다음과 같은 결론에 도달할 수 있었다. 첫째, 제안된 구조학습 기반 뉴로-퍼지 예측시스템의 성능이 MLP, RBFN, 랜덤하게 초기화한 뉴로-퍼지 모델의 성능보다 우수하다. 특히, MAPE의 평균뿐만 아니라 MAPE의 표준 편차가 다른 방법과 비교해서 매우 향상되었다. 이는 제안된 방법으로 신뢰성이 향상된 예측시스템을 구현할 수 있음을 의미한다. 둘째, 제안된 방법에서 초기 구조 백크를 설계할 때 사용한 학습 데이터가 과거 1년간의 전력 수요 데이터를 포함하기 때문에 설계된 초기 구조 백크는 1, 2, 3, 4월에 대해서 그 성능이 계속 유지된다. 물론 7, 8월과 같은 여름철에 대해서 실험해 보진 않았지만, 일단 한번 구조 학습을 통해 초기 구조 백크가 설정되면, 적어도 향후 1년간 유효할 것이라 판단된다. 셋째, 계절별로는 겨울 (1월과 2월)이 봄 (3월과 4월)보다 예측이 어렵고, 요일별로는 월요일과 일요일이 다른 요일보다 예측이 어려우며, 시간별로는 출퇴근시간에 대한 예측이 어렵기 때문에 MAPE의 평균과 표준편차 모두 크게 나오게 된다. 이러한 현상은 학습 데이터 50개를 사용한 MLP 기반 예측시스템과 RBFN 기반 예측시스템 모두에서 관찰되었다.

5. 결 론

본 논문에서는 MLP의 단점을 극복하기 위한 효과적인 대안으로 인식되고 있는 뉴로-퍼지 모델을 이용하여 한 시간 앞의 전력 수요를 예측하기 위한 체계적인 방법을 제안하였다. 특히, 한 시간마다 뉴로-퍼지 모델을 학습할 때 구조 학습을 반복해야 하는 문제를 해결하기 위해서 4개의 요일 유형에 대해 시간별 초기 구조를 미리 학습하고, 이를 초기 구조 백크에 저장하는 방법을 제안하였다. 제안된 방법에서는 초기 구조 백크에서 예측 시점에 해당되는 초기 구조를 선택하여 뉴로-퍼지 모델을 초기화하기 때문에, 초기화된 뉴로-퍼지 모델의 인자 학습 과정만을 매 시간 반복하게 된다.

제안된 방법의 효용성을 검증하기 위해서 1993년과 1994년 한국 전력의 실제 데이터를 이용하여 모의 실험을 수행

하였다. 모의 실험 결과 제안된 방법은 MAPE의 평균이 1% 수준이었고 MAPE의 표준편차는 평균적으로 1.06 정도였다. 한시간 앞의 전력 수요 예측시스템에서 실용화가 가능한 MAPE의 평균이 보통 1% 미만인 점을 고려하면, 본 논문에서 제안한 방법이 실용화 측면에서도 만족할 만한 성능을 얻는다는 것을 알 수 있었다.

제안된 방법의 성능을 기존의 방법과 비교하기 위해서 MLP를 이용한 예측시스템과 RBFN을 이용한 예측시스템을 구현하여 동등한 조건에서 성능의 비교 평가를 수행하였다. 제안된 방법의 성능이 MAPE의 평균과 MAPE의 표준편차 측면에서 모두 우수한 결과를 보였으며, 이로부터 제안된 방법이 보다 정확하고 신뢰할 수 있는 예측시스템을 구현할 수 있다고 결론지을 수 있었다.

향후 진행되어야 할 연구의 방향은 크게 세 가지로 요약된다. 첫째, 본 논문에서는 다루지 않았지만, 전력 수요 예측 시스템 개발에서 매우 중요한 고려 사항인 계절 (온도) 요인과 특수일에 대한 연구를 수행할 예정이다. 특히, 계절별 요인에 대해 향상된 예측시스템을 구성하기 위해서, 초기 구조를 학습할 때 1년치의 학습 데이터를 모두 사용하지 않고, 계절별로 제한된 학습 데이터만을 이용하여 계절별 초기 구조를 구성하는 방법을 연구할 예정이다. 이 경우 계절별로 초기 구조 백그를 재학습하는 문제를 해결해야 한다. 둘째, 입력 변수 선정에 관한 연구를 진행할 계획이다. 본 논문에서는 14개의 입력 변수를 사용하였는데, 너무 많은 입력 변수를 갖는 뉴로-퍼지 모델에서는 학습된 퍼지 규칙을 해석하는 것이 어렵게 된다. 이러한 문제를 극복하기 위해서 다수의 후보 입력 변수에서 유용한 소수의 입력 변수를 선정하는 작업이 필수적이다. GMDH (Group Method for Data Handling)와 같은 방법이 이 문제 해결에 유용할 것이라 판단된다 [22, 23]. 마지막으로 예측 문제의 범위를 확장할 계획이다. 제안된 방법은 1시간 앞의 전력 수요를 예측하는 문제뿐만 아니라, 다음날 24시간 전체의 전력 수요를 예측하는 문제 또는 향후 일주일 168시간의 전력 수요를 예측하는 문제 등과 같이 다양한 수요 예측 문제에서도 매우 효과적인 것으로 기대된다.

감사의 글

본 연구가 가능하도록 전력 수요 데이터를 제공해 준 한국전력에 감사드립니다. 또한 본 논문의 심사를 위해 애써주신 심사위원께 진심으로 감사드립니다. 본 연구는 강릉대학교 기성회 학술연구비(신진교수 과제)의 지원으로 수행되었으며 이에 감사드립니다

참 고 문 헌

[1] 김형수, 문경준, 황기현, 박준호, 이화석, "신경회로망을 이용한 단기전력부하 예측용 시스템 개발," 대한전기학회 하계학술대회 논문집, pp. 24-26, 1999.

[2] D. C. Park, M. El-Sharkawi, R. Marks, A. Atlas, and M. Damborg, "Electrical Load Forecasting Using an Artificial Neural Network," *IEEE Trans. on Power Systems*, Vol. 6, No. 2, pp. 442-449, May 1991.

[3] T. W. Chow and C. T. Leung, "Neural Network Based Short-Term Load Forecasting Using Weather Compensation," *IEEE Trans. on Power Systems*, Vol. 11, No. 4, pp. 1736-1742, Nov. 1996.

[4] K. H. Kim, J. K. Park, K. J. Hwang, and S. H. Kim, "Implementation of Hybrid Short-Term Load Forecasting System Using Artificial Neural Networks and Fuzzy Expert Systems," *IEEE Trans. on Power Systems*, Vol. 10, No. 3, pp. 1534-1539, Aug. 1995.

[5] A. Khotanzad, R. C. Hwang, A. Abaye, and D. Maratukulam, "An Adaptive Modular Artificial Neural Network Hourly Load Forecaster and its Implementation at Electric Utilities," *IEEE Trans. on Power Systems*, Vol. 10, No. 3, pp. 1716-1722, Aug. 1995.

[6] A. G. Bakirtzis, J. B. Theocharis, S. J. Kiartzis, and K. J. Satsios, "Short Term Load Forecasting Using Fuzzy Neural Networks," *IEEE Trans. on Power Systems*, Vol. 10, No. 3, pp. 1518-1524, Aug. 1995.

[7] H. Mori and H. Kobayashi, "Optimal Fuzzy Inference for Short-Term Load Forecasting," *IEEE Trans. on Power Systems*, Vol. 11, No. 1, pp. 390-396, Feb. 1996.

[8] M. Sugeno and T. Yasukawa, "A Fuzzy Logic Based Approach to Qualitative Modeling," *IEEE Trans. Fuzzy Systems*, Vol. 1, No. 1, pp. 7-31, Feb. 1993.

[9] B. H. Wang and H. J. Cho, "Structure Identification of Neuro-Fuzzy Models Using Genetic Algorithms," submitted to *IEEE Trans. Fuzzy Systems*, 1998.

[10] M. Kubat, "Decision Trees can Initialize Radial Basis Function Networks," *IEEE Trans. Neural Networks*, Vol. 9, No. 5, pp. 813-821, Sept. 1998.

[11] C. T. Lin and C. S. G. Lee, "Neural-Network-Based-Fuzzy Logic Control and Decision System," *IEEE Trans Comput.*, Vol. 40, No. 12, pp. 1320-1336, Dec. 1991.

[12] J. S. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Trans. Syst., Man, Cyberm.*, Vol. SMC-23, No. 3, pp. 665-684, May/June, 1993.

[13] J. S. R. Jang and C. T. Sun, "Neuro-Fuzzy Modeling and Control," in *Proc of IEEE*, Vol. 83, No. 3, pp. 378-406, Mar. 1995.

[14] L. X. Wang and J. M. Mendel, "Generating Fuzzy Rules by Learning from Example," *IEEE Trans. Syst., Man, Cyberm.*, Vol. SMC-22, No. 6, pp. 1414-1427, 1992.

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

[16] H. Takagi and N. Suzuki, T. Koda, and Y. Kojima, "Neural Networks Designed on Approximate

- Reasoning Architecture and their Applications," IEEE Trans. Neural Networks, Vol. 3, No. 5, pp. 752-760, 1992.
- [17] K. B. Cho and B. H. Wang, "Radial Basis Function Based Adaptive Fuzzy Systems and their Applications to System Identification and Prediction," Fuzzy Sets and Systems, Vol. 83, pp. 325-339, 1996.
- [18] P. Wasserman, Advanced Methods in Neural Computing, Van Nostrand Reinhold, 1993.
- [19] J. R. Quinlan, C4.5: Programs for Machine Learning, San Mateo: Morgan Kaufmann, 1993.
- [20] L. Wang, R. Langari, and J. Yen, "Principal Components, B-splines, and Fuzzy System Reduction," in Proc. Fuzzy Logic for the Applications to Complex Systems, W. Chiang and J. Lee, Ed. World Scientific, 1995.
- [21] J. Moody and C. Darken, "Fast Learning in Networks of Locally-Tuned Processing Units," Neural Computation, Vol. 1, pp. 281-294, 1989.
- [22] A. G. Ivakhnenko, "Polynomial Theory of Complex Systems," IEEE Trans. Syst., Man, Cyberm., Vol. SMC-12, pp. 364-378, 1971.
- [23] S. Farlow, ed., Self Organizing Methods in Modeling : GMDH-Type Algorithm, Marcel Dekker, Inc., New York, 1984.

저 자 소 개



박 영 진 (朴 永 鎭)

1972년 11월 16일 생. 1998년 강릉대 전자공학과 졸업(학사). 2000년 동 대학원 전자공학과 졸업(석사). 현재 (주) 옵토마인 재직

Tel : 0342-604-2100, Fax : 0342-604-101

E-mail : yjpark@optomine.com



왕 보 현 (王 普 賢)

1964년 9월 10일 생. 1987년 연세대 전기공학과 졸업. 1989년 Georgia Institute of Technology 전기공학 석사. 1991년 Georgia Institute of Technology 전기공학 (공박) 1991-1998년 LG 종합기술원 책임연구원. 1998-현재 강릉대학교 전자공학과 조교수

Tel : 0391-640-2384, Fax : 0391-640-2244

E-Mail : bhw@kangnung.ac.kr



심 현 정 (沈 炫 正)

1976년 12월 15일 생. 1999년 강릉대 전자공학과 졸업(학사). 2000년 현재 동 대학원 전자공학과 석사과정. 관심분야 : 인공 신경망, 예측 시스템, 데이터 마이닝

E-mail : shimmy@kangnung.ac.kr