

## 신경회로망 앙상블을 이용한 부하추종운전시의 원자로 모델링

### Modeling of Nuclear Reactor in Load Following Operations with Neural Network Ensemble

이상경, 이은철

서울대학교

서울시 관악구 신림동 산 56-1

장진욱, 성승환

한국원자력연구소

대전광역시 유성구 덕진동 150

#### 요 약

신경회로망 앙상블을 이용하여 부하추종운전시 제어봉과 보론의 변화에 대해 원자로 출력, 제논반응도가, Axial Offset을 연소도에 따라 예측할 수 있는 모델을 만들었다. 신경회로망 학습방법으로는 Scaled Conjugate Gradient 알고리즘을 사용하여 학습 성능을 증가시켰고 ONED94 코드를 통해 학습 자료를 구했다. 고리 1호기 9주기 일일부하추종운전을 모사해 본 결과에 의하면 단일 신경회로망으로 구성된 모델보다 예측 성능이 개선되었고 추가되는 자료에 대한 학습이 어려운 문제도 해결하였다.

#### Abstract

Neural network model is used for prediction of reactor power, Xenon worth and axial offset with different burnups in load following operations. Scaled Conjugate Gradient algorithm is employed to speed up the training of neural networks that consist of neural network ensemble. Training and testing data are generated by ONED94 code. The test results presented exhibit the neural network ensemble model is more effective than single neural network model for the prediction and can be prone to the problem of incremental learning.

## 1. 서론

시스템의 입출력 관계를 기반으로 하는 신경회로망은 시변특성, 부정확한 모델 등에 대한 실시간 적응보상이 가능하고 임의의 비선형 시스템을 식별할 수 있는 범용 함수 근사자 역할을 할 수 있음이 밝혀져 있기 때문에 비선형 특성을 갖고 모델링이 어려운 원자력 발전소의 시스템에 효율적으로 사용될 수 있다.

출력 변화에 따른 원자로의 상태를 신경회로망으로 예측하는 것은 빠르고 정확한 계산 모델이 필요한 모델예측제어기(Model Predictive Controller)에 적용할 수 있고 운전원이 운전지침을 판단하는 데 중요한 역할을 할 수 있다. 최근 이와 관련된 연구들이 진행되고 있는데 Seong은 가압경수로형 원자로에서 제어봉과 보론을 사용하여 이로 인한 원자로 출력, A.O(Axial Offset), 반응도 의 변화를 신경회로망으로 계산하였다.[1] 그러나 현재 시간에서 만들어질 수 있는 원자로의 상태들에 대해 다양한 학습자료를 준비해야하고 신경회로망의 입력을 구성해야 하는 단점이 있다. 이에 반해 Mehrdad는 원자로를 시간연속 시스템 관점으로 보고 현재까지 오는 과거의 정보들로부터 다음 시간 간격에서의 결과를 예측하는 신경회로망 모델을 만들었다.[2] 이는 현재 상태의 많은 주변 정보들을 이용하기보다 과거 정보를 이용하여 미래를 예측하는 방식이므로 부하추종운전같이 시간에 따른 원자로의 상태를 모사하는 데 적합한 방법이라 생각된다. 하지만 이 논문에서는 오류 역전파(Error Backpropagation) 알고리즘을 사용하여 학습성능이 떨어지고 부하추종운전시에 사용되는 보론을 고려하지 않았다.

이러한 두 논문에서의 문제점을 개선하여 제어봉과 보론의 변화에 따른 원자로 출력, A.O, 제논반응도가(Xe Worth)를 예측하는 신경회로망 모델을 개발하였다.[3] 그러나 이 모델도 원자로의 연소도에 따른 변화를 반영하지 못하였다. 연소도까지 고려한 자료를 적용하게 되면 자료의 양이 증가함에 따라 신경회로망의 학습 성능이 떨어진다. 또한 신경회로망을 적용하여 모델을 만들 때 처음부터 완벽한 학습자료가 주어지기는 현실적으로 불가능하고 주어진다고 하더라도 상당히 많은 자료를 모두 섭렵하도록 학습하기는 불가능할 것이다. 다시 말하여 학습자료 부족으로 인한 외삽의 문제가 발생할 것이고 기존에 학습한 신경회로망에 추가로 주어진 자료를 통한 학습이 이루어지면 과거에 학습했던 것은 망각하게되는 문제가 발생할 것이다. 이러한 문제를 해결하기 위한 방안으로 신경회로망 앙상블이 제시되고 있다.[4] 신경회로망 앙상블은 독자적으로 훈련된 다수의 신경회로망을 결합한 시스템으로 이는 다르게 훈련된 각각의 신경회로망들이 오차공간 상에서 서로 다른 지역 국소(Local Minima)에 빠지게 되고 이렇게 지역 국소에 빠진 상태에서의 판단을 잘 결합하면 전체적인 성능은 단일 시스템에 비해 훨씬 더 향상되리라는 것이다. 하지만 효과적인 앙상블의 구축을 위해서는 앙상블 멤버들 간의 상관관계가 아주 낮아야 하며 또한 각 앙상블 멤버들은 전체 문제를 어느 정도는 정확하게 학습하면서도 서로들간에 불일치 하는 부분이 존재해야 한다. 신경회로망의 구조(노드의 수, 연결개수 등)를 어떻게 설정할 것이며, 각 신경회로망의 출력 값으로부터 최적의 결합 성능을 보이는 방법

을 어떻게 선택할 것인지가 앙상블의 일반화 성능을 결정 짓는 중요한 요소이다.

본 논문에서는 제어봉과 보론의 변화에 따른 원자로 출력, A.O, 제논반응도가(Xe Worth)를 계산할 때 연소도도 고려한 원자로 모델링을 하기 위해 신경회로망 앙상블을 적용하였다. 신경회로망 앙상블을 구성하는 단일 신경회로망의 구성은 기존 연구[3]에서 이루어진 NARX 신경회로망을 채택하였으며 학습방법은 Scaled Conjugate Gradient 알고리즘을 사용하였다. 이 알고리즘은 오류역전파 알고리즘이 오차함수의 일차 근사를 사용하고 사용자 정의를 필요로하는 파라미터들로 인해 학습성능이 떨어지는 문제와 오차함수의 2차근사를 사용하는 학습방법이 Hessian 행렬로 인한 계산 시간과 메모리의 증가 문제를 극복할 수 있다. 학습 및 검증을 위한 데이터를 만들기 위해 ONED94 코드로 고리 1호기 9주기에서의 일일부하추종운전을 수행하였다. 학습 자료는 2-6-2-14의 시간에 대해 5일동안 100-50-100, 100-60-100, 100-70-100, 100-80-100, 100-90-100 의 출력변화를 주어 6가지의 연소도에 따른 자료를 사용하였고 검증 자료는 제논이 부하추종운전에 주는 영향이 크기 때문에 다른 제논의 거동을 보이도록 2-4-2-16 형태의 자료를 사용하였고 외삽성능을 보기 위해 학습하지 않은 연소도 구간에서의 자료를 사용하였다.

## 2. 원자로 신경회로망 모델

### 1) 단일 신경회로망 구조[3]

제어봉과 보론의 농도를 변화시켰을 때 원자로 출력, 제논반응도가, A.O 의 값이 어떻게 되는지를 예측하기 위해 사용한 신경회로망의 구조는 그림 1과 같이 입력부분, 1개의 은닉층 그리고 출력층의 구조를 가지고 있다. 은닉층은 입력노드수의 2배에 해당하는 노드로 구성하였다. 신경회로망의 입력으로서는 현재(t)와 과거의 제어봉위치, 보론농도, 신경회로망모델의 출력인 원자로출력, A.O, 제논반응도가의 과거 값을 신경회로망 모델의 입력으로 케환하여 사용하며 출력은 현재 다음시간간격(t+1)의 원자로출력, A.O, 제논반응도가로 구성한다.[그림 2] 신경회로망의 입출력 관계를 표시하면 다음과 같다.

$$(P(t+1), Xe(t+1), A.O(t+1)) = F [Xe(t), \dots, Xe(t-q), P(t), \dots, P(t-q), A.O(t), \dots, A.O(t-q), CR(t), \dots, CR(t-p), BC(t), \dots, BC(t-p)]$$

여기서 P는 원자로출력, Xe는 제논반응도가, A.O는 Axial offset, CR은 제어봉위치, BC는 보론농도를 의미한다. 또한 p,와 q 는 시간지연단위를 나타내며 시스템의 차수를 사용한다.

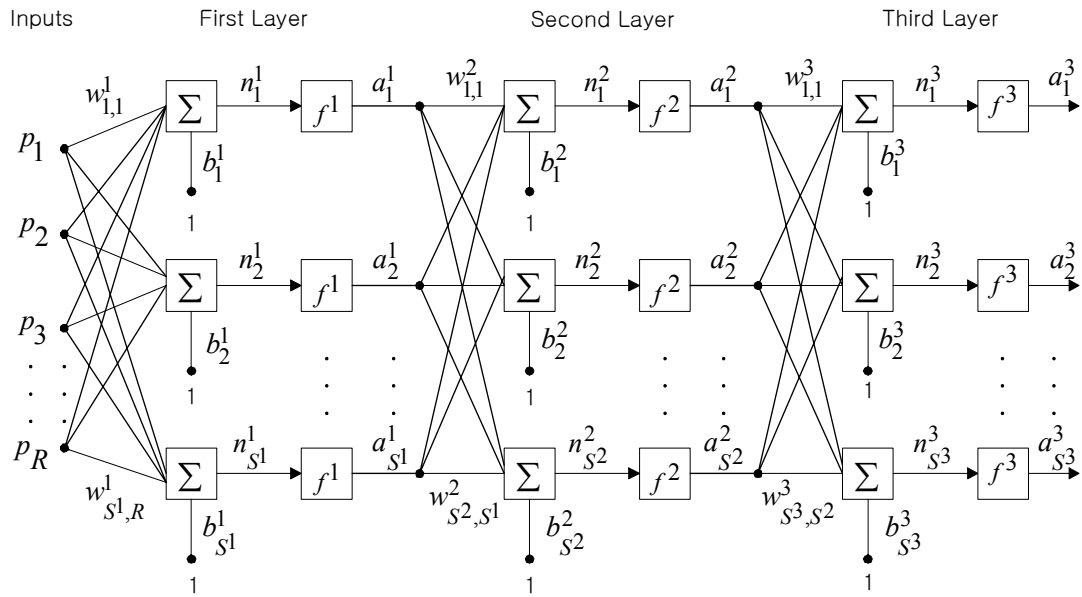


그림 1. 다층 퍼셉트론(Multi Layer Perceptron)

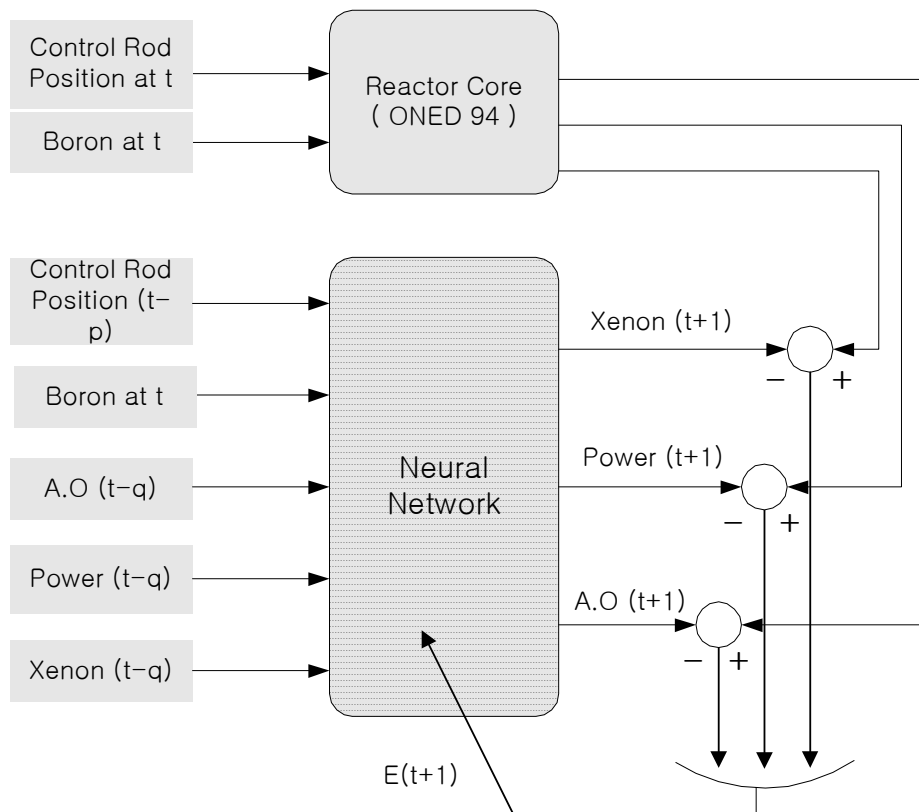


그림 2. 노심 모사를 위한 신경회로망 구조

이와 같이 다층신경회로망 구조에서 입력으로서 시간 지연된 값들과 신경회로망의 출력 값을 다시 궤환하여 시간 지연된 값을 사용하는 모델을 NARX 신경회로망이라고 말한다. NARX 신경회로망은 장,단기적 역학이 동시에 나타나는 비선형 시스템을 모사하는데 있어 효과적이다.

## 2) 학습알고리즘

기존에 신경회로망을 이용하여 원자료를 모사하였던 논문들은 기본적인 오류역전과 알고리즘으로 학습방법을 사용하거나.[1]-[2] 이것보다 학습능력과 계산속도가 빠른 방법이지만 계산 시간과 메모리 문제가 발생할 수 있는 Levenberg Marquardt 알고리즘을 사용하였다.[3] 본 논문에서는 앙상블을 구성하는 각각의 단일 신경회로망을 학습하는데 Scaled Conjugate Gradient 알고리즘[5]을 사용하여 이러한 문제점을 극복하였다. 오차함수(E)를 2차로 근사하여 사용할 때 나타나는 Hessian 행렬은 2차미분에 의한 계산시간과 메모리 문제뿐만 아니라 singular 이거나 ill-condition 인 경우엔 최적해(w)를 구할수 없기 때문에 Scaled Conjugate Gradient 알고리즘은 Hessian 행렬을 1차 미분 근사를 하였고, positive definite가 되도록 보정하는 방법을 사용하였다. 자세한 계산 과정은 그림 3에 나타나 있다.

## 3) 신경회로망 앙상블( Neural Network Ensemble )

앙상블의 중요한 이론적 배경중의 하나는 bias-Variance dilemma 이다.  $F_i(\mathbf{x}, D)$ 를 i번째 네트워크의 실제 출력 값이라 하면, 앙상블 출력은 다음과 같이 표현된다.

$$F(\mathbf{x}, D) = \frac{1}{M} \sum_{i=1}^M F_i(\mathbf{x}, D)$$

위의 식에서 M은 네트워크의 수를 의미하고  $\mathbf{x}(\in R^n)$  은 입력패턴을, D는 훈련 데이터의 집합을 의미한다. 실제 원하는 출력 값을 d 라 하면 결합된 앙상블 시스템의 평균 제곱 오차의 기대값(Expected Mean Squared Error)은 아래와 같이 각각의 네트워크에 대한 항으로 표현이 가능하다.

$$\begin{aligned} & E_D[(E[d | \mathbf{x}] - F(\mathbf{x}, D))^2] \\ &= (E_D[F(\mathbf{x}, D)] - E[d | \mathbf{x}])^2 \\ &+ E_D\left[\frac{1}{M^2} \sum_{i=1}^M (F_i(\mathbf{x}, D) - E_D[F_i(\mathbf{x}, D)])^2\right] \\ &+ E_D\left[\frac{1}{M^2} \sum_{i=1}^M \sum_{j \neq i} (F_i(\mathbf{x}, D) - E_D[F_i(\mathbf{x}, D)]) \right. \\ &\quad \left. \times (F_j(\mathbf{x}, D) - E_D[F_j(\mathbf{x}, D)])\right] \end{aligned}$$

|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| STEP 1 | Choose weight vector $\tilde{w}$ and scalars $0 < \sigma \leq 10^{-4}$ $0 < \lambda_1 \leq 10^{-6}$ $\bar{\lambda}_1 = 0$<br>set $\tilde{p}_1 = \tilde{r}_1 = -E(\tilde{w}_1)$ $k = 1$ and success = true                                                                                                                                                                                                                                                                                                                                                                      |
| STEP 2 | If success = true, then calculate 2nd order information<br>$\sigma_k = \sigma /  \tilde{p}_k $ $\tilde{s}_k = (E(\tilde{w}_k + \sigma_k \tilde{p}_k) - E(\tilde{w}_k)) / \sigma_k$ $\delta_k = \tilde{p}_k^T \tilde{s}_k$                                                                                                                                                                                                                                                                                                                                                      |
| STEP 3 | Scale $\delta_k$ : $\delta_k = \delta_k + (\lambda_k - \bar{\lambda}_k)  \tilde{p}_k ^2$                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| STEP 4 | If $\delta_k \leq 0$ then make the Hessian matrix positive definite<br>$\bar{\lambda}_k = 2(\lambda_k - \delta_k /  \tilde{p}_k ^2)$ $\delta_k = -\delta_k + \lambda_k  \tilde{p}_k ^2$ $\lambda_k = \bar{\lambda}_k$                                                                                                                                                                                                                                                                                                                                                          |
| STEP 5 | Calculate step size : $\mu_k = \tilde{p}_k^T \tilde{r}_k$ $\alpha_k = \mu_k / \delta_k$                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| STEP 6 | Calculate the comparison parameter<br>$\Delta_k = 2\delta_k [E(\tilde{w}_k) - E(\tilde{w}_k + \alpha_k \tilde{p}_k)] / \mu_k^2$                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| STEP 7 | If $\Delta_k \geq 0$ then<br>$\tilde{w}_{k+1} = \tilde{w}_k + \alpha_k \tilde{p}_k$ $\tilde{r}_{k+1} = -E(\tilde{w}_{k+1})$ $\bar{\lambda}_k = 0$ <i>success = true</i><br>If $k \bmod N = 0$ then restart algorithm : $\tilde{p}_{k+1} = \tilde{r}_{k+1}$<br>else : $\beta_k = ( \tilde{r}_{k+1} ^2 - \tilde{r}_{k+1}^T \tilde{r}_k) / \mu_k$<br>$\tilde{p}_{k+1} = \tilde{r}_{k+1} + \beta_k \tilde{p}_k$<br>If $\Delta_k \geq 0.75$ , then reduce the scale parameter $\lambda_k = \frac{1}{4} \lambda_k$<br>else : $\bar{\lambda}_k = \lambda_k$<br><i>success = false</i> |
| STEP 8 | If $\Delta_k < 0.25$ , then increase the scale parameter<br>$\lambda_k = \lambda_k + (\delta_k (1 - \Delta_k) /  \tilde{p}_k ^2)$                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| STEP 9 | If the steepest descent direction $\tilde{r}_k \neq 0$ then set $k = k + 1$ and go to step 2 else terminate and return $\tilde{w}_{k+1}$ as the desired value                                                                                                                                                                                                                                                                                                                                                                                                                  |

그림 3. 신경회로망 학습을 위한 Scaled Conjugate Gradient 알고리즘

이 식에서 첫 번째 항은 결합된 시스템의 전체 편향(bias)을 의미하고, 두 번째 와 세 번째 항은 각각 멤버들의 출력에 대한 분산(variance)과 공분산(Covariance)에 해당된다. 편향과 분산 문제는 모든 학습알고리즘에서 고유의 문제이므로 앙상블 시스템에서 전체 결합 성능에 가장 영향을 미치는 것은 공분산이라고 할 수 있다. 앙상블 시스템이 가장 좋은 일반화 성능을 내기 위해서는 앙상블의 각 멤버 네트워크들 간에 공분산이 아주 낮아야 함을 이 식을 통해서 알 수 있다. 즉, 앙상블을 구성하는 단일 신경회로망들이 상관관계가 아주 낮아야하며 전체문제를 어느정도는 정확하게 학습하면서도 서로간에 불일치하는 부분이 존재해야 한다. 이를 위해 초기 가중치를 다르게 주거나, 구조를 달리하거나, 중첩되지 않는 학습자료를 분배하는 등의 노력을 한다. 본 논문에서는 다양한 연소도에 따른 학습자료로 각각의 단일 신경회로망을 학습하도록 하였다. 이렇게 각각 학습된 단일 신경회로망들의 출력값들로부터 최종 출력값을 계산하는 방법은 Simple Averaging 과 Weighted Averaging을 주로 사용한다. 각각의 신경회로망에서 나온 출력 값에 weight 값을 곱한 후 더해서 최종 앙상블의 결과 값을 구해낸다. 이때 각 출력에 곱해지는 weight값을 구하는 것 또한 최적화 문제이므로 유전자알고리즘 등 다양한 연구들이 진행되고 있으나 본 논문에서는 다음과 같이 사용하였다.

$$W_i = \frac{1 - E_i}{\sum_k (1 - E_k)}$$

여기서,  $E_i$  값은 각 신경회로망을 학습시켰을 때의 MSE(Mean Squared Error)로 사용하였다. 따라서 최종 앙상블의 출력  $Y$ 는  $\sum W_i y_i$  가 된다.

원자로 모델링에 사용되는 앙상블의 구조는 그림 4에 나타나 있고, 학습을 위해 고리 1호기 9주기에서 6가지의 연소도에 따른 자료를 구하여 사용하였으므로 6개의 단일 신경회로망이 결합되어 앙상블을 이루었다. 새로운 자료가 주어질때마다 이를 학습한 단일 신경회로망을 추가하면 된다.

### 3. 신경회로망 모델의 학습 및 검증

노심의 거동을 모사 계산하는데 사용한 ONED94 코드[6]는 Nodal방법을 사용하는 1차원 유사과도상태 코드로서 현재 국내 Westinghouse형 원전의 운전 지원용으로 사용되고 있는 ONED90을 CE형 노심에서도 사용할 수 있도록 개선한 것이다. 이 코드에 필요한 단면적 자료는 DIT-ROCS 체계로 생산하였으며 특정 노심 연소도에 대해 ROCS 계산치를 기준으로 Tuning 절차를 거친 후 사용한다.

ONED94 코드를 통해 고리 1호기 9주기에서 6가지의 연소도(224, 485.38, 746.76, 1008.14, 1269.52, 1530.9 MWD/MTU)에 따라 5일동안 2-6-2-14의 시간으로 100-50-100,

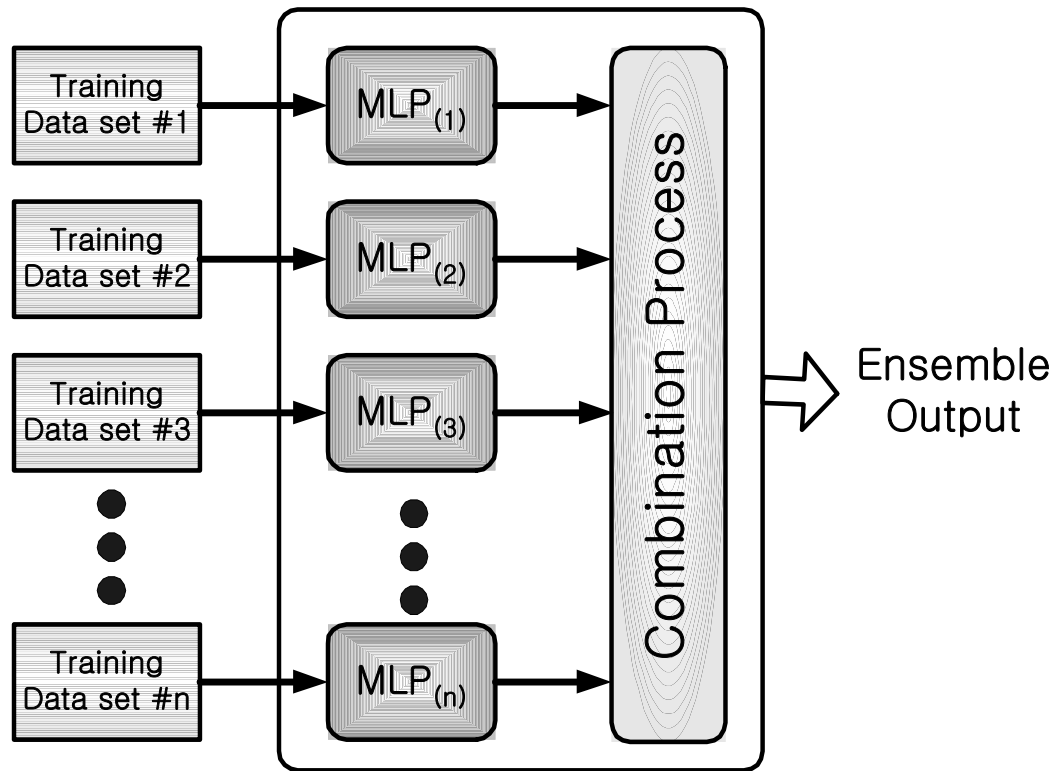


그림 4. 신경회로망 앙상블 구조

100-60-100, 100-70-100, 100-80-100, 100-90-100의 출력변화를 주어 운전한 자료를 얻었다. 이때 제논은 천이상태로 하였고 데이터 획득시간 간격은 5분을 사용했다. 이때 구한 6가지의 연소도에 따른 자료들을 6개의 단일 신경회로망에 각각 학습하였다.

실제 원자력발전소 자료를 활용할 때는 시간에 따른 제논반응도가의 정보는 없다. 그러나 코드를 통해 값을 구하여 학습을 해놓으면 제어봉과 보론의 변화에 따른 거동은 예측할 수 있는 것을 검증단계에서 볼 수 있다.

앙상블을 구성하는 단일 신경회로망 모델에서  $p$ 는 2,  $q$ 는 5를 사용하였고 입력의 개수는 총 24개, 은닉층의 노드 수는 48, 출력의 개수는 3 이다. 신경회로망의 학습은 Scaled Conjugate Gradient 알고리즘을 이용하여 Mean Squared Error가  $10^{-5}$  이하가 될 때까지 반복하였으나 iteration 횟수는 500회 이하로 제한 하였다.

앙상블의 학습하지 않은 연소도에 대한 성능검증을 위하여 연소도가 1792.28 MWD/MTU 일때의 자료를 모사해 보았다. 학습을 위해 생성된 자료와 같이 2-6-2-14의 시간에 대해 5일동안 100-50-100, 100-60-100, 100-70-100, 100-80-100, 100-90-100의 출력변화를 주어 운전하였다.[그림 6] 비교를 위해 6가지의 연소도 자료를 앙상블이 아닌 하나의 신경회로망으로 학습한 모델에서도 계산하였다.[그림 5] 그림에서 ASI(Axial



Shape Index)는 A.O 값과 부호만 반대이다. 6종류의 연소도 값을 사용하였기 때문에 하나의 신경회로망으로 학습한 것도 오차가 큰 편은 아니지만 앞에서 언급하였듯이 자료의 양이 많거나 추가될 때마다 하나의 신경회로망으로 학습하는 데에 문제가 발생하게 된다. 외삽의 결과를 보면 앙상블이 하나의 신경회로망으로 학습하는 것보다 오차가 적게 나옴을 알 수 있다.

실제로 제논의 영향이 부하추종운전을 어렵게 하기 때문에 제논의 변화가 있을 때에도 잘 모사할 수 있는지 검증해 보았다. 자료는 연소도 300 MWD/MTU에서 2-4-2-16의 시간에 대해 5일동안 100-50-100, 100-60-100, 100-70-100, 100-80-100, 100-90-100의 출력 변화를 주어 운전하여 얻었다. 그림 7은 하나의 신경회로망으로 학습하였던 모델에서의 결과이고 그림 8은 앙상블 모델에서의 결과이다. 여기에서 앙상블의 성능이 단일 신경회로망보다 좋으며 학습하지 않은 형태의 운전상황에 대해서도 제대로 모사할 수 있음을 알 수 있다.

#### 4. 결론

많은 자료와 추가적으로 학습자료가 주어질 때 새로 행해지는 학습으로 인하여 과거 학습한 자료에 대해서도 좋은 결과를 가질 수 있는 신경회로망 앙상블을 이용하여 원자력 발전소의 부하추종운전시 연소도를 고려하여 제어봉과 보론의 변화에 따른 원자로 출력, A.O, 제논반응도가를 예측하는 모델을 개발하였다.

고리1호기 9주기에서 6가지의 연소도에 대한 학습자료로 앙상블을 구성하는 각각의 단일 신경회로망을 학습하였고 출력을 변화시키는 운전 형태는 같지만 학습하지 않은 연소도에서 모사한 경우와 학습한 연소도 범위내에서 학습하지 않은 운전 형태를 모사한 경우에 대해 검증을 하였다. 단일 신경회로망으로 구성한 모델과 비교한 결과 신경회로망 앙상블로 구성한 모델이 모두 우수한 성능을 보였다.

본 논문에서 개발한 모델은 앙상블을 구성하여는 각각의 신경회로망을 서로 다른 연소도에 대해 학습하였기 때문에 연소가 이루어질 때마다 주어지는 자료만 학습하여 추가하면 되는 장점이 있다. 개발된 모델은 Model Predictive Control에서 상태를 예측하는데 필요한 모듈로서 사용할 수 있고 다른 시스템의 신경회로망 모델링과 연계하여 운전원 지원시스템으로서의 모사기를 구성하는데 활용 가능할 것이다.

#### 참고문헌

1. S.H. Seong, et.al, "Application of neural networks to analyze load-follow operation in a pressurized water reactor." *Nuclear Technology*, vol.128, pp. 276-283, 1999.

2. Mehrdad B., et.al, "Identification of a nuclear reactor core(VVER) using recurrent neural network." *Annals of Nuclear Energy*, vol.29, pp.1225-1240, 2002.
3. 이상경 외, " NARX 신경회로망을 이용한 원자로 부하추종운전 모사," 한국원자력학회 2003 춘계학술발표회논문집, 2003년 5월.
4. A.Sharkey(Ed.), Combining artificial neural Nets: Ensemble and Modular Multi-net Systems, Springer, London, 1999.
5. Moller M., "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Networks*, vol.6, pp.525-533, 1993.
6. KAERI, *User's manual ACE/ONED*, 1996.

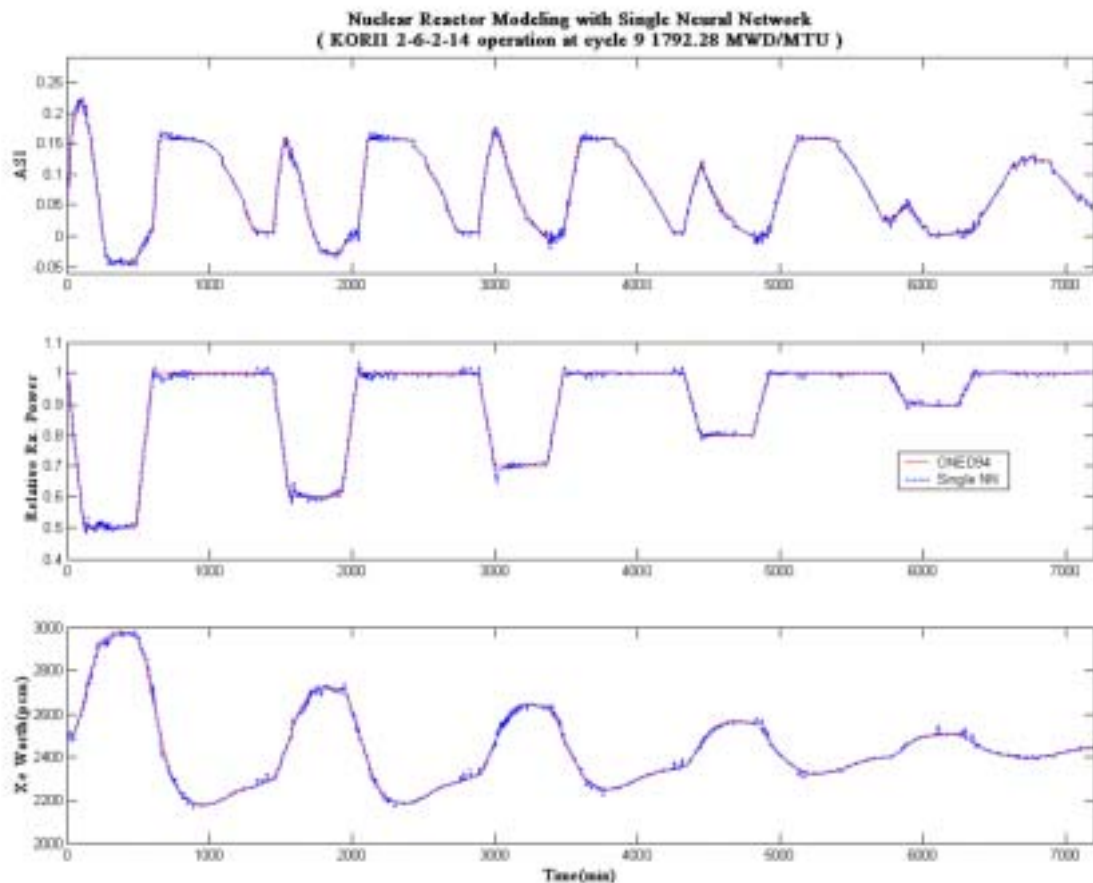


그림 5. 단일 신경회로망 모델의 검증결과 ( 1792.28 MWD/MTU )

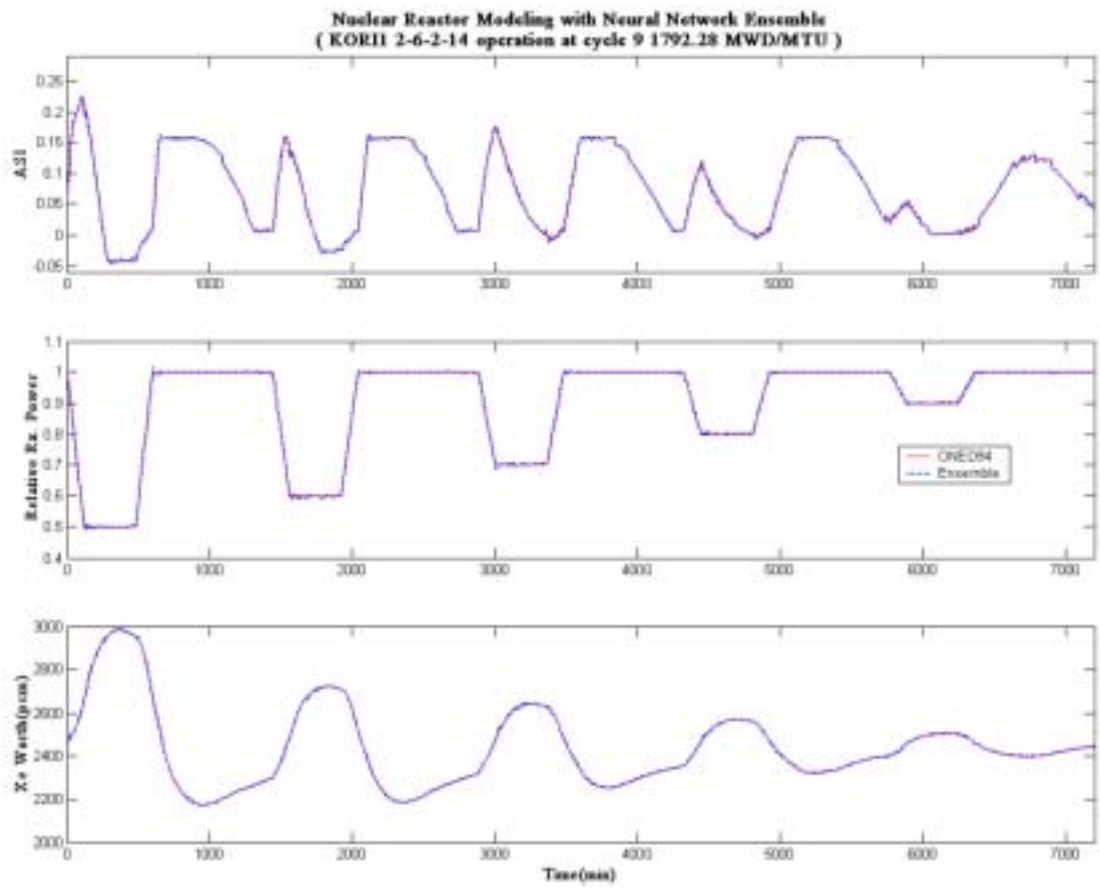


그림 6. 신경회로망 앙상블 모델의 검증결과( 1792.28 MWD/MTU )

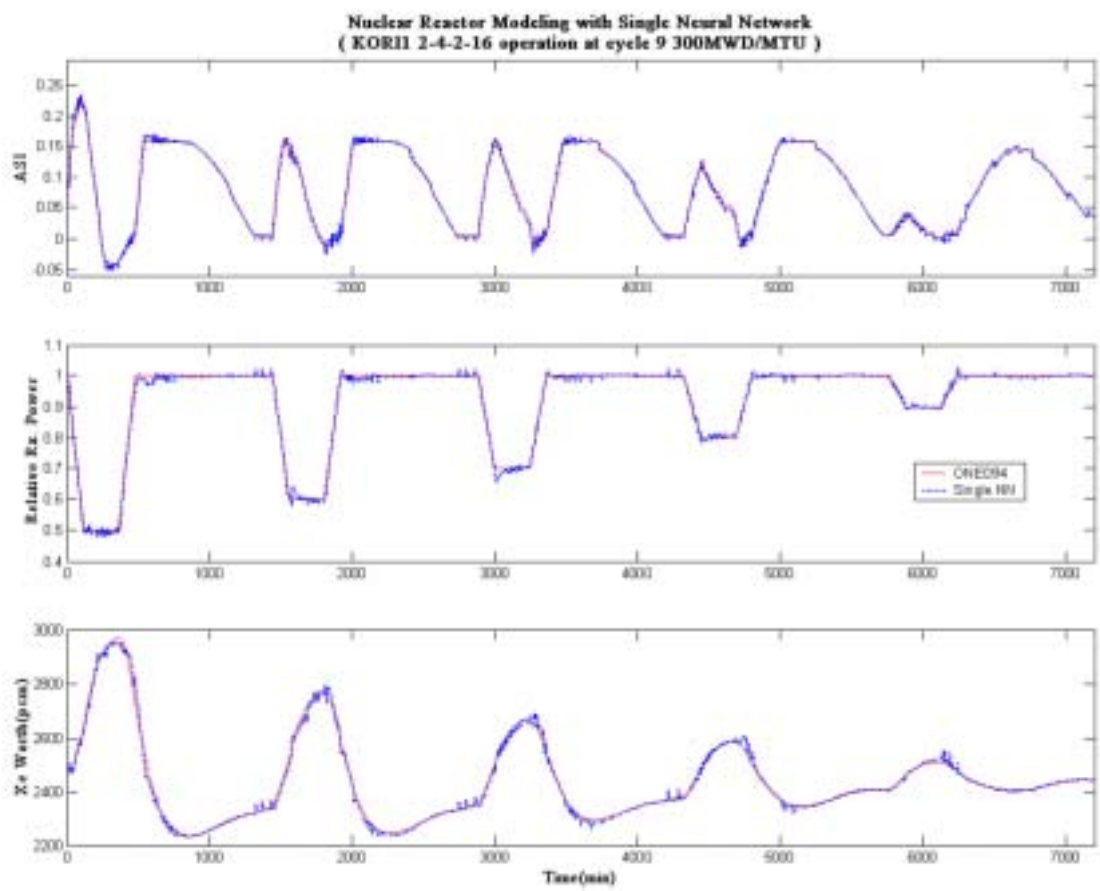


그림 7. 단일 신경회로망 모델의 검증 결과( 300MWD/MTU )

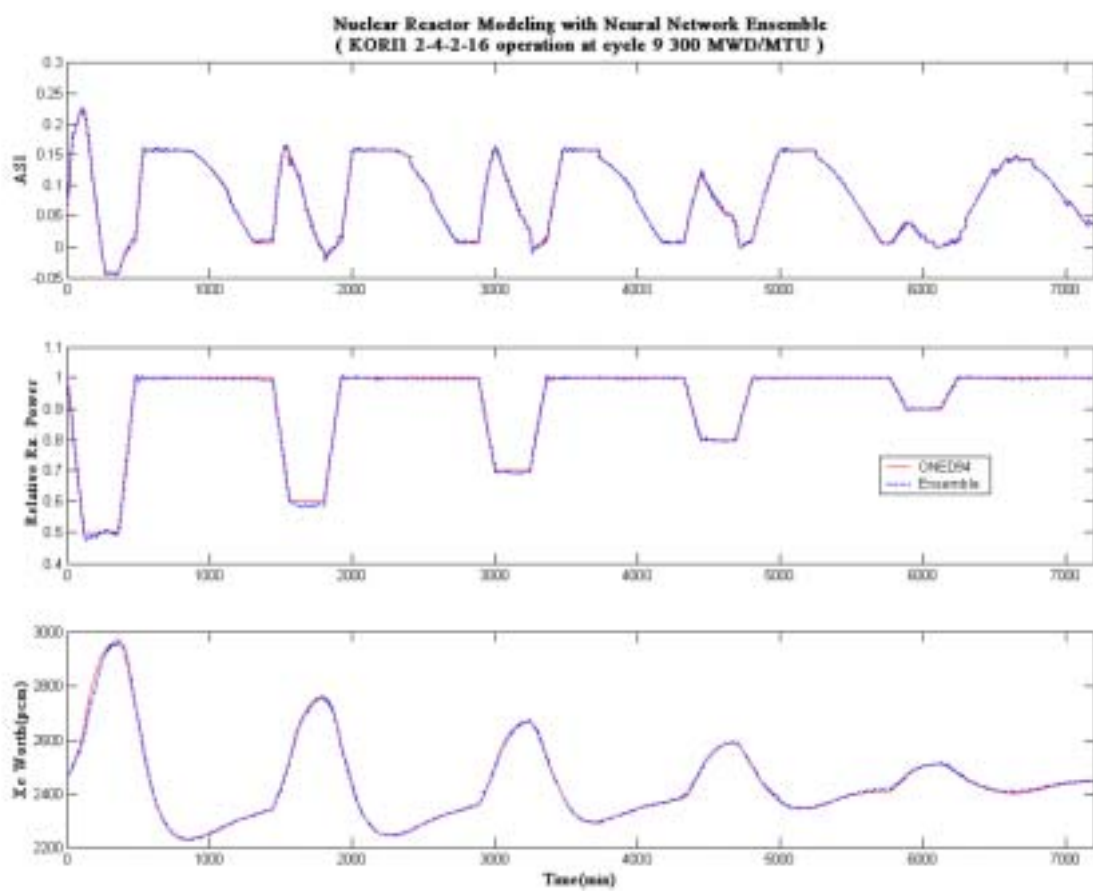


그림 8. 신경회로망 앙상블 모델의 검증 결과( 300MWD/MTU )