A MODIFIED ELMAN RECURRENT NEURAL NETWORK (MERNN) TO PREDICT DNBR RATIO IN A PRESSURIZED HEAVY WATER REACTOR

Masoud Farzam, Faramarz Moattar

Dept. of Electrical and Computer Eng., McMaster University, Hamilton, ON, Canada

ABSTRACT

So far, few researches have been carried out on the estimation of DNBR ratio in a nuclear reactor. However, they were mainly static approaches and so have not considered the temporal changes due to the dynamics of reactor and the correlation of major parameters in the reactor. In this paper a modified elman recurrent neural network (MERNN) is developed to predict and capture the temporal changes of DNBR ratio, paired with a reactor core simulator which simulates the neutronic and thermo-hydraulic dynamics of a 40MW pressurized heavy water research reactor. The reactor simulator is coupled to the MERNN and they interchange data in every time step resulting to a prediction of DNBR ratio rather than static estimation in the previous methods. COBRA III-C code is used for data acquisition and generalization tests but the critical heat flux routine has been modified to be used for our special low pressure heavy water case. Several surveys has been done to achieve the best correlation for the heavy water properties. Results show a good prediction in many transients varies from multiple failure of heat exchanger and pressurizer as well as loca accident to a simple positive reactivity insertion. Modified rule which is applied to the back propagation through time algorithm shows a fast and improved training and memorization which can be used for the other online modeling of the real time complex and nonlinear systems.

1. INTRODUCTION

Departure from nucleate boiling ratio (DNBR) is one of the most critical parameters in safety issues of a nuclear reactor. A safe design of a reactor should guaranty a minimum DNBR of 1.3. A good prediction of such important parameter could be helpful in preventing of hazard accidents in a nuclear plants. DNBR is a function of reactor power, pressure, inlet core temperature and radial peaking factor. However these factors are correlated to each other due to the dynamics of reactor. A recurrent neural network should be used for capturing of this complex nonlinear dynamics. In this paper we propose a modified elman recurrent neural network which is coupled to a core dynamics simulator to predict the transients of DNBR. It is obvious that any operator action which is followed by a minimum DNBR prediction of less than 1.3 can be prevented by safety protection system consequently. This paper is organized as follow. First we explain a little about the reactor configuration and then we describe the method used for the low pressure heavy water critical heat flux estimation. The rest of the paper introduces an elman recurrent neural network structure as well describing the modified back propagation through time algorithm. Finally the training and generalization of MERNN will be considered and the prediction for several transients are tested. Noise consideration and loss of data tests are carried out as well.

2. REACTOR CONFIGURATION

Our considered reactor is of the vessel type with uranium oxide fuel and zircalloy clad. The moderator and coolant are both heavy water. The fuels are located in the fuel assemblies and there are totally 150 fuel assemblies with 18 fuel pins in each. The normal coolant pressure is 2 bar. Coolant enters to core at 50°C and exits at 70°C. Maximum produced thermal power in the core is 40 MW. Fuel pin diameter is 1.363 cm and fuel active length is 343 cm. For thermo-hydraulic sub-channel analysis, we select 1/12 sector of total fuel assembly section and divide it in to five sub-channels as figure 1:

![Figure 1: 1/12 section of symmetry of the fuel assembly layout](image)

The subchannel geometry data are as follows:
we know that:

\[ q^u = q^u_{max} \cos(\frac{\theta}{\Theta}) \quad Q_{total} = \int_{-1/2}^{1/2} \cos(\frac{\theta}{\Theta}) Pdz \]

we know that:

\[ Q_{total} = q^u_{ave} PL \]
\[ q^u_{ave} = q^u_{max} \frac{H}{L} \left( \sin \left( \frac{H}{2} \right) - \delta_2 \right) + \sin \left( \frac{H}{2} - \delta_1 \right) \]

\[ \delta_1, \delta_2, H, L, P, \] are respectively total and active fuel length and fuel perimeter and extrapolated lengths. A final result can be achieved as follows:

\[ q^u = q^u_{max} \frac{H}{L} \left( \sin \left( \frac{H}{2} \right) - \delta_2 \right) + \sin \left( \frac{H}{2} - \delta_1 \right) \]

According to above equation, table 1 can be extracted for relative axial heat flux distribution along the core.

Table 1 is used as an input to the COBRA III-C code. We should modify the CHF routine in order to be accurate for the low pressure case as well as for the heavy water coolant type.

3. CHF ROUTINE MODIFICATION

In [7] has been proposed to use the following formula for the calculation of local heat flux in the vertical round tubes:

\[ (q_{max})_{cr,z} = \frac{A + B \Delta h_{sub,i}}{C + z} \]

If we want to be more accurate we should consider the nonuniformity in heating along the tube. So we substitute the following modified formula in the case of non-uniform heating:

\[ (q_{max})_{cr,z} = \frac{A + B \Delta h_{sub,i}}{C f(z) + \int_0^z f(z)dz} \]

\( f(z) \) is heat flux profile, \( \Delta h_{sub,i} \) is the inlet subcooling and \( z \) the distance along the tube. The parameters \( A, B, C \)

are modified for non circular channels and for an annuli they are proposed as follows:

\[ A = 67.4D_h^{0.38} (m/s) 10^{-6} \cdot 10^{0.102} \cdot (1 - 0.744e^{[6.512D_h(m/s) 10^{-6}])}} \]
\[ B = 0.258D_h^{1.261} (m/s) 10^{-6} \cdot 0.817 \]
\[ c = 185.0D_h^{1.415} (m/s) 10^{-6} \cdot 0.212 \]

\( m^r \) is mass flow rate, \( D_r \) and \( D_h \) are respectively hydraulic and heated equivalent diameter. The above correlation covers the following ranges:

\[ m^r : 190 - 8430 k g/m^2 s \quad \Delta h_{sub,i} : 0 - 958 k J/kg \]

The rms error of this correlation is 5.9% and for pressure other than 69 bar, the parameter \( A \) should be modified to \( \Delta h_{v} \) where the \( \Delta h_{v} \) is the latent heat of vaporization at the pressure considered in Btu/lb.

4. SIMULATION OF THE CORE

We now go through the dynamic simulation of the core. This simulator is used for prediction of reactor power in case of any change in the reactivity or core inlet temperature or pressure as well as coolant flow. We first simulate the core neutronic. We use a point kinetics model as follows:

\[ \frac{dn}{dt} = \frac{\delta n - \beta n + \lambda c}{\lambda} \quad \frac{dc}{dt} = \frac{\beta c - \lambda c}{\lambda} \]

where \( n, c, \lambda, \beta, \rho \) are respectively neutron and precursor density and effective precursor decay constant, effective prompt neutron life time, fraction of delayed neutrons and reactivity. For computational purposes it is convenient to use equivalent normalized versions of the above equations:

\[ \frac{dn_r}{dt} = \frac{\delta n_r - \beta n_r + \lambda c_r}{\lambda} \quad \frac{dc_r}{dt} = \frac{\beta c_r - \lambda c_r}{\lambda} \]

\( n_0, c_0 \) are initial equilibrium for neutron and precursor densities. Using the normalized point kinetics equations the reactor power can be represented as:

\[ P_a(t) = P_0 a_i n_i(t) \]

where \( P_a, P_0 \) are respectively the instant and initial reactor power in MW. Now we use the early works done by Schultz [1] to present a two temperature feedback mechanism for a reactor. First the time dependent heat transfer from fuel to coolant and the net heat removal from the coolant are:

\[ P_c(t) = h_f(T_f - T_c) \quad P_c(t) = M(T_i - T_c) \]

where \( P_c, P_c, T_f, T_r, T_i, T_c \) are respectively, power transferred from fuel to coolant, average...
reactor fuel temperature, average reactor coolant temperature, outlet temperature, inlet temperature. And also the $h_f$ is the heat transfer coefficient between fuel and coolant, and $M$ is the mass flow rate times heat capacity of the water.

The differential equation for lumped fuel and coolant temperature are then:

$$f_f P_a(t) = \mu_f \frac{dT_f}{dt} + P_c(t)$$

$$(1 - f_f) P_a(t) + P_c(t) = \mu_c \frac{dT_c}{dt} + P_c(t)$$

where $f_f$, $\mu_f$, $\mu_c$ are respectively fraction of reactor power deposited in the fuel, total heat capacity of the fuel and structural material and coolant. We use the reactivity feedback equations represented by Robert Edward [1]:

$$\delta\rho = \delta\rho_r + \alpha_f(T_f - T_0) + \alpha_c(T_c - T_0)$$

where $\delta\rho_r$, $\alpha_f$, $\alpha_c$, $T_0$, $T_0$ are reactivity due to the control rods, fuel and coolant temperature reactivity coefficients and equilibrium temperatures.

Now we should give an impression for $h_f$ which depends on the flow regimes. We consider a sub cooled nucleate boiling regimes. In [2] Chen and Thom have proposed correlations for this case. Here we use Thom’s correlation to calculate the heat transfer coefficient.

$$T_W - T_{sat} = 22.659(q^*0.5e^{-p/787})$$

where $p$ is the pressure of coolant in bar and $q^*$ is the surface heat flux in $MW/m^2$. $T_W$, $T_{sat}$ are temperatures at fuel surface and saturation temperature at desired pressure. Now we need the correlations for thermo hydraulic properties of heavy water. McGee, Glen and Ji Zhang also Garland [5] has done many works in this regard. But here we use a complete job done by Ahmet Durmaz [3] in early 1997 for developing approximation functions for heavy water properties. Here we just introduce those functions that we use for our simulator:

### 4.1. Approximation function for thermohydraulic properties of heavy water

We first present the correlation for the saturation temperature and pressure $h_f$, $h_g$ as follows:

$$T_{sat} = a + c P_{sat}^{d/2}$$

$$P_{sat} = \left( \frac{T_{sat} - a}{c} \right)^{1/e}$$

$$h_f = \sum_{i=0}^{4} a_i P_{sat}^{e} + f ln(P_{sat})$$

$$h_g = \sum_{i=0}^{4} a_i (P_{sat} - b)^{e} + c(P_{sat} - d)^{e}$$

In the [3], A complete table has been presented to show the values for parameters of $a, b, c, e$ in every desired range of pressure and temperature. In Table 2 we just present the values that are matched to our range of interest ($P=2$ bar, $Tavg=60^\circ C$).

Experimental results have shown that the above functions are quite accurate with a maximum error of 2% desired range. Now we introduce the designed structure for the neural network.

### 5. NEURAL NETWORK ARCHITECTURE

We employ a recurrent network structure to capture the transient state of the DNB ratio as a function of any change to the power, flow, pressure or inlet temperature in the core. Since we have already implemented a simulator for transients of such parameters in the core, then we have a temporal system at the input layer and so we would have a variable output and these all lead to the prediction of the DNB ratio. Previous research has mainly concentrated on the estimation of the DNB ratio at a fixed input parameters condition. However it is obvious that dynamics of reactor forces us to use a temporal network rather than a static one. Since the reactor itself is a nonlinear and complex system with many inherent feedbacks, we would obviously consider these feedbacks by selection of a recurrent neural network. In [2], Horne and Giles show that RMLP networks give an optimum and powerful representation for the nonlinear systems comparing to the TDNN structure. Here we use an elman recurrent network since we do not need the feedback from output to the previous layers (as it is in th Jordan structure) and only hidden layers feedbacks are feed to the system. We consider four neurons for the input layer to be matched to four input parameters of the reactor core. Then we consider a hidden layer consisting of 6 neurons in it. There are feedbacks from input layer and hidden layer which are denoted as in the figure 2. Here a context layer is added to the structure which retains information between observation. At each time step new inputs are fed in to the MERNN. The previous content of the hidden layer are passed in to the context layer. These then feedback to the hidden layer in the next time step. This modification improves the memorization capability of the network and so can be useful for higher order complex dynamic system.
The values of \( a \) and \( b \) are computed recursively:

\[
e^y(i) = \delta^y(i) = e(i)
\]

\[
e^b(i) = W^y\delta^y(i) + I_n\delta^e(i + 1)
\]

\[
\delta^b(i) = e^b(i) \otimes \phi'(s(i))
\]

\[
e^e(i) = \alpha I_n \delta^e(i + 1) + W^e\delta^b(i)
\]

\[
\delta^e(i) = e^e(i)
\]

The process is initialized at time \( k \) according to:

\[
e^y(k) = \delta^y(k) = e(k) = y(k) - y^b(k)
\]

\[
\delta^b(k) = e^b(k) \otimes \phi'(s(k))
\]

\[
e^e(k) = W^e\delta^b(k)
\]

The symbol \( \otimes \) denotes the multiplication element by element, \( I_n \) is an identity matrix and \( \phi'(\cdot) \) is the derivative of the hyperbolic tangent function.

### 5.1. training algorithm

The algorithm is actually a back propagation through time procedure done by Werbos (1990) to be adapted to the new proposed MERNN structure. The gradient is approximated considering a finite number of \( n \) of previous sampling periods. So we define it as:

\[
E(k) = \sum_{i=k-n}^{k} e(i)^2, \ e(k) = y(k) - y^b(k)
\]

where \( e(k) \) is the modeling error at time step \( k \).

Suppose \( x, y, u \) be respectively indexes for the weight matrices of context-hidden, input, output layers and \( \rho, \mu \) be respectively the momentum and learning rate. For updating these weights we have:

\[
\Delta W^i(k + 1) = \rho_m \Delta W^i(k) - \mu_m (1 - \rho_m) \frac{\partial E(k)}{\partial W^i(k)}
\]

The recurrent network is expanded into a multilayer feed forward network, being a new layer added at each time step. The computation of derivatives in above equation has been performed by Rumelhart and Williams (1986)[4]:

\[
\frac{\partial E(k)}{\partial W} = \sum_{i=k-n}^{k} \delta^b(i) x_e(i)^T
\]

\[
\frac{\partial E(k)}{\partial W} = \sum_{i=k-n}^{k} \delta^b(i) u(i - 1)^T
\]

\[
\frac{\partial E(k)}{\partial W} = \sum_{i=k-n}^{k} \delta^b(i) x_b(i)^T
\]

The values of and are computed recursively:

<table>
<thead>
<tr>
<th>parameter</th>
<th>a0</th>
<th>a1</th>
<th>a2</th>
<th>a3</th>
<th>a4</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 sat</td>
<td>-87.6554</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>97.6839</td>
<td>-</td>
<td>0.1424</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>5.2 sat</td>
<td>-45.005</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>61.556</td>
<td>-</td>
<td>0.1877</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>h_f</td>
<td>405.974</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>431.976</td>
<td>-</td>
<td>0.1375</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>h_g</td>
<td>11597.974</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>13890.0</td>
<td>5.0</td>
<td>0.002951</td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: table of parametric data for heavy water correlations

Figure 2: Elman recurrent ANN for predicting the DNBR

5.2. Data acquisition

Temporal neural networks are trained by introducing a transient state in the input layer. Normally this transient should consider all or most of the possible output response of the system. For this reason we have considered a mix and complex transient which is produced by considering increasing and decreasing of the reactivity in the core followed by the loca accident as well as the increasing the pressure and temperature due to the pressurizer and heat exchanger rapture. This actually covers many possible ways to the decreasing of the DNBR and so losing the safety features. Figure 3 show the reactivity, pressure, flow and inlet temperature changes which are considered for the training set. The power transient then is obtained by running our developed simulator and so due the dynamic of the reactor we can get the result as shown in the figure 3.

A batch file is then written to be used as an input file to the COBRA III-C thermo hydraulic code to obtain the transient state of DBNR ratio. Following the above procedure, we sampled every 1/4 second of total 450 second transient curve and so 1800 sample were provided. The first 1500 are used for the network training and the remained are used for generalization test too. There are two other transients that are used for generalization test. Training has reached to the desired goal in 7014 epochs and the final error is 9.98E-6.
<table>
<thead>
<tr>
<th>net.Lw(2,1)</th>
<th>net.Lw(2,2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1305</td>
<td>-0.1300</td>
</tr>
<tr>
<td>0.5238</td>
<td>0.3820</td>
</tr>
<tr>
<td>-0.3656</td>
<td>-0.5257</td>
</tr>
<tr>
<td>0.8770</td>
<td>-0.8190</td>
</tr>
<tr>
<td>-0.2118</td>
<td>0.3002</td>
</tr>
<tr>
<td>-0.4989</td>
<td>0.4965</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>net.iw(1,1)</th>
<th>net.Lw(1,1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.0017</td>
<td>-1.9166</td>
</tr>
<tr>
<td>1.7140</td>
<td>1.3899</td>
</tr>
<tr>
<td>-1.3501</td>
<td>1.9407</td>
</tr>
<tr>
<td>1.5677</td>
<td>-0.8863</td>
</tr>
<tr>
<td>net.Lw(3,2)</td>
<td>0.8983</td>
</tr>
<tr>
<td>net.b(2,1)</td>
<td>1.3827</td>
</tr>
<tr>
<td>net.b(1,1)</td>
<td>-1.0246</td>
</tr>
<tr>
<td>net.b(3,1)</td>
<td>-0.4330</td>
</tr>
</tbody>
</table>

Table 3: Table of the weights and biases for the MERNN network

The weight and bias matrixes are shown in table 3,4:

Figure 3: temperature, flow, pressure changes and simulator result of output power

Figure 4: Predicted and calculated values for DNBR; Left) case 1 Right) case 2

Figure 5: Predicted and calculated values for DNBR; Left) case 2 Right) case 1

6. TEST RESULTS

The first transient is due to the increasing of the normalized reactivity from -0.0285 to -0.0283 at first 100 sec and then decreasing to -0.291 in the next 350 sec. The pressure and temperature are increasing due to the pressurizer and heat exchanger rapture and the flow is decreasing due to the local accident. The reactivity change and the simulation results due to these transients is shown in the figure 3. Figure 4 shows the predicted and calculated values for the DNBR by the MERNN network and the COBRA III-C code. It can be seen that for the first 1500 sample training goal is achieved very well and for the next 300 test sample the predicted and calculated values are close together with the same direction.

Figure 5 shows the transient for an increase in the normalized reactivity from -0.283 to -0.274 in the first 300 sec and then decreasing to -0.0278 in the next 150 sec. The change in the other input parameters are the same as case 1. The simulation results is shown in the 5. This case also was being tested by the MERNN network and the result of prediction by MERNN and calculation by COBRA III-C are shown in Figure 4. It can be seen that the prediction is done very well and exactly in the same direction. The maximum error for this case is 7overestimation comparing to the COBRA III-C calculation, but it can be seen that the transient memorization has been done very well. Also the prediction delay is too low and this shows that the network is too fast.
6.1. Noise considerations

To test the noise tolerant capability of the network we have considered a train pulse reactivity change which is shown in the 6. The power transient resulted from the simulation shows an exponential modulated curve. We have considered that pressure and flow and also inlet temperature sensors are noisy so they show a random variation around a constant value as shown in 6. The neural network output is shown in 6. It can be seen that the MERNN network has done a good prediction and it has followed the original noiseless transients with an acceptable error. There is a considerable error in discontinuous points and this is because the MERNN network has no time to follow this very fast changes and also we have not trained this network for this fast jumps cases. So the MERNN structure can be used as a fault tolerant predictor system for the prediction of DNBR ratio as well.

7. CONCLUSION

[hb] We developed a modified Elman neural network for the prediction of DNBR ratio in a pressurized heavy water reactor. The tests show that this structure is able to predict the DNBR in many cases with an acceptable error. The MERNN structure follows the change directions during the transients very fast and shows also a good memorization of the network. The structure is tolerant to noise as well as loss of data coming from sensors. So this structure can be used as a part of safety protection system to prevent from any wrong action done by the operators resulting to a DNBR less than

1.3. The MERNN network coupled with the developed core simulator can be used also for prediction of severe accidents like LOCA and heatexchanger or pressurizer rapture.

8. REFERENCES


