Abstract

Objective: Artificial Neural Networks became a powerful tool for dynamic modelling of non linear physical systems and for prediction of specific parameters of complex systems. This work aims to model a heat exchanger using different neural networks and to investigate their performance in predicting its outlet temperature.

Methods: In this work four different neural networks namely Elman Recurrent Neural Networks (ERNN), Time Delay Neural Networks (TDNN), Cascade Feed Forward Neural Networks (CFFNN) and Feed Forward Neural Networks (FFNN) are modelled for the prediction of the outlet liquid temperature of a saturated steam heat exchanger from its liquid flow rate. A benchmark dataset consisting of 4000 tuples is used to train, validate and test the performance of each neural network model. Findings: All the four ANN models are trained, validated and tested which predicted the outlet temperature of the Heat exchanger with acceptable accuracy and the Elman Recurrent Neural Network is found to have the best accuracy by having the lowest Mean Square Error (MSE) and best Regression due to its feedback connections. Applications: The Artificial Neural Network Models simulated, especially the Elman Neural Networks are good at prediction of operational parameters of a physical system and hence can be used in prediction and forecasting of operational parameters of engineering, medical, financial and environmental systems.

Keywords: Artificial Neural Networks, Elman Recurrent Neural Network, Heat Exchanger, Prediction

1. Introduction

Dynamic neural networks are good at modelling of non linear systems. Sen M et al applied neural networks to predict the dynamic behaviour and steady state of the heat exchangers. They established that using dynamic neural networks for modelling thermal process offer a reliable and fast way of prediction of their performance and the models can be updated continuously. Neural Networks were used for the prediction of heat transfer rates of shell and tube heat exchangers by Wang et al. Patra et al. used the artificial neural networks for modeling the Intermediate Heat Exchanger (IHX) subsystem in nuclear reactor. Wind resource assessment and forecasting systems for forecasting wind speed and direction were proposed by Rotich et al. in which Feed Forward, Cascade Feed Forward and Elman Networks were used for modelling. Russo, et al. designed and implemented a neural network approach for prediction of temperatures of groundwater heat pump systems. The hybrid artificial neural network algorithms and k-nearest neighbor algorithms are used in blogger classification process. A control strategy which employs neural network predictive and fuzzy controller with an auxiliary manipulated variable was developed by Vasickaninova and Bakosova for tubular heat exchanger used for pre-heating of petroleum by hot water. An online performance monitoring system for shell and tube heat exchanger was developed using Artificial Neural Networks (ANNs) by Ahilan et al. Further neural networks have been used in thermal systems for performance prediction and dynamic control and heat transfer analysis. A comparative study of Feed Forward, Cascaded Forward, Elman networks and Self Organizing Map neural networks for intrusion detection.

*Author for correspondence
system was conducted by Nazir. Mummadisetty et al. applied neural networks for the prediction of solar radiation. They used various neural networks namely MLP neural networks, Cascade Feed forward Neural Networks (CFNN) and Elman Recurrent Neural Networks (ERNN). A hybrid method using decision tree and feed forward network is used in medical application for coronary heart disease risk prediction by Akila and Chandramathi. A hybrid fuzzy Jordan network is simulated and used for efficient and robust intrusion detection system by Dhivya and Sivanandam. An ANN approach is implemented for Enhancing the Performance of MANETS by Sumathi and Sundaram.

Motivated by the literature four types of Neural networks namely Elman Recurrent Neural Network (ERNN), Feed Forward Neural Network (FFNN), Time Delay Neural Network (TDNN) and Cascade Feed Forward Neural Network (CFFNN) are employed to model the heat exchanger to predict the outlet liquid temperature of a saturated steam heat exchanger from its liquid flow rate. The networks are trained, tested and their performance were analyzed and compared.

2. System Modelling for the Predictor

The heat exchanger process is considered to be a significant benchmark example for nonlinear control design purpose. The data set is a bench mark data set contributed by Bittanti, MILANO (Italy) which is obtained from Database for the Identification of Systems (DaISy), Department of Electrical Engineering, ESAT/SISTA, Leuven and Belgium. In this liquid-saturated steam heat exchanger process, pressurized saturated steam flowing through a copper tube heats the water. The outlet liquid temperature (th) is the output. The liquid flow rate (q) is the input. The inlet liquid temperature and the steam temperature are held at constant values.

The Data set consists of

a. Exchanger Inputs - An array of 4000 scalar values representing 4000 time steps of liquid flow rates (q)
b. Exchanger Targets - An array of 4000 scalar values representing 4000 time steps of outlet liquid temperatures (th)

The sample time for this data was one second (1 sec). Out of the 4000 tuples 60% of the data are used for Training, 20% are used for validation and the remaining 20% are used for testing. The simulations are carried out with four different network models namely ERNN (Elman Recurrent Neural Network), FFNN (Feed Forward Neural Network), TDNN (Time delay Neural Network) and CFFNN (Cascade Feed Forward Neural Network) using the same data set and the same performance evaluation criteria.

The Mean Square Error (MSE) and the Regression are used as the evaluation criteria. Mean Square Error (MSE) is given by:

\[
mSE = \frac{\sum (Actual\ Value - Predicted\ Value)^2}{n} = \frac{\sum (Error)^2}{n}
\]

2.1 Artificial Neural Networks

Due to the non-algorithmic nature and the adaptive learning capability the artificial neural networks are considered as good computational models. The information is iteratively processed by training the networks using the input and output relationship of the past data.

2.1.1 Feed Forward Neural Networks (FFNN)

Feed Forward Neural Networks also called as Multi-Layer Perception Neural Network have an input layer, hidden layer (one or more) and an output layer consisting of non-linearly-activating neurons. Each layer is connected to the next layer in a feed forward way. Figure 1 shows a Feed Forward Neural Network.

In our work the sigmoid function is used as the activation function for the input-hidden layer and a linear function is used for hidden-output layer. A standard back propagation algorithm is used in FFNN for supervised learning.

![Figure 1. Feed forward neural network.](image)
2.1.2 Cascade Feed Forward Neural Networks (CFFNN)

Cascade-forward networks include a connection from the input and every previous layer to following layers\textsuperscript{17,18} making them capable of learning high complexity associations\textsuperscript{11}. In CFFNN every input layer is connected to every other hidden layer in the cascade form. The CFFNN model is illustrated in Figure 2.

2.1.3 Elman Recurrent Neural Networks (ERNN)

ELMAN neural network has an input layer, one or more hidden layer and a recurrent layer. The recurrent layer presents one step delay to the hidden layer. The recurrent link layer stores the detailed information of the hidden layer and this is found to retain the memory\textsuperscript{18}. The ERNN is illustrated in Figure 3.

![Figure 2. Cascade feed forward neural network.](image)

![Figure 3. Elman Recurrent Neural Network with context layer\textsuperscript{11}.](image)

The ELMAN neural network is trained based on current input along with the past state output\textsuperscript{11}. Elman neural networks are best suited for prediction related problems because of the recurrent connections.

2.1.4 Time Delay Neural Networks (TDNN)

Time Delay Neural Network (TDNN) is a network architecture with a tap delay line associated with its input. Time delay network enables the network to have an effective dynamic response to time series data\textsuperscript{21}.

2.2 The Architecture

As the data has one input parameter (liquid flow rate) the input layer has one neuron. Number of Hidden layer neurons play a vital role in the performance. If more number of neurons is present, then the error calculations may consume more time and the convergence may be slower. If less number of neurons is present, then the accuracy is lost. There is no Hard and fast rule or formula to fix the count of Hidden layer Neurons\textsuperscript{20}.

In this work Number of Hidden Neurons are set as 3, 5, 10 and 12 and the performance is analyzed. The best results are obtained at 10 hidden layer neurons. There are different training algorithms like “Levenberg-Marquardt” (trainlm), Bayesian regression” (trainbr). Gradient descent” (traindx), one step secant back propagation” (trainoss) etc., The Bayesian Regression (trainbr) is found to be an effective after several trail runs with different algorithms. Table 1 illustrates the architecture using which all the four neural networks are modelled.

2.3 Algorithm

The following steps are performed to carry out the prediction process. This procedure is repeated for all the four networks.

### Table 1. The Architecture for modelling

<table>
<thead>
<tr>
<th>Input Neurons</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Neurons</td>
<td>1</td>
</tr>
<tr>
<td>No of Hidden Layers</td>
<td>1</td>
</tr>
<tr>
<td>Hidden layer Neurons</td>
<td>10</td>
</tr>
<tr>
<td>Training Algorithm</td>
<td>TrainBr - Bayesian Regularization Back Propagation</td>
</tr>
<tr>
<td>Activation Function</td>
<td>Tan Sigmoid for input-hidden layer Linear function for hidden-output layer</td>
</tr>
<tr>
<td>Performance</td>
<td>MSE</td>
</tr>
</tbody>
</table>
1. Initialize weights for input to hidden nodes and for hidden to output nodes.
2. Present training samples \((x, t)\) to the neural network.
3. For each hidden unit calculate the output and present it to the output unit.
4. For each output unit calculate the output value.
   Compare with the target value and compute the error.
5. If the error doesn't meet the stopping criteria (specified error limit), update the weights and go to step 2.
6. If the error meets the stopping criteria (error limit specified) the training is completed.
7. Present the test data to the trained network and measure the performance metrics.

Using this procedure each neural network is trained, validated and the outputs are arrived. Figure 4 shows the flow diagram of the back propagation algorithm.

### 3. Simulation Results

All the four Neural Networks considered for this work are modelled, trained, validated and tested for the prediction of the outlet liquid temperature of a liquid-saturated steam heat exchanger taking its liquid flow rate as input. The comparisons of the performance of the different Neural Networks are shown in Table 2.

The error plots are shown for all the networks from Figures 5a to 5d.

#### Table 2. Performance evaluation of different neural networks

<table>
<thead>
<tr>
<th>Neural Network</th>
<th>MSE</th>
<th>Regression R</th>
<th>Epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDNN</td>
<td>1.2289</td>
<td>0.75112</td>
<td>896</td>
</tr>
<tr>
<td>ELMAN</td>
<td>0.0059</td>
<td>0.99989</td>
<td>848</td>
</tr>
<tr>
<td>CFFNN</td>
<td>1.8948</td>
<td>0.56768</td>
<td>816</td>
</tr>
<tr>
<td>FFNN</td>
<td>1.8911</td>
<td>0.56811</td>
<td>910</td>
</tr>
</tbody>
</table>

![Figure 4.](image-url) Flow diagram showing the algorithm.

![Figure 5.](image-url) (a) Error plot of TDNN, (b) Error plot of ERNN, (c) Error plot of CFFNN, (d) Error plot of FFNN.
The output temperature plots are shown for all the networks from Figures 6a to 6d.

The regression plots for all the networks are shown from Figures 7a to 7d.

4. Performance Analysis and Discussion

The Error Plots, Output temperature Plots and regression Plots for all the Networks are analyzed. Studying the Output Temperature plots, it is clear that all the Networks have been trained and can predict the outlet temperature. But their performance varies. From Table 2 it is clear that the Elman Recurrent Neural Network has the lowest value of MSE = 0.0059 which confirms the best accuracy of prediction process.

Figure 6. (a) Output plot of TDNN, (b) Output plot of ERNN, (c) Output plot of CFFNN, (d) Output plot of FFNN.
Based on the error plots it is evident that ERNN has the lowest error range between -0.15 and 0.2, whereas the other neural networks have the error range between -5 and +4. The regression coefficient of ERNN is $R = 0.99899$ which indicates that the output variable and the input variable are better related and the output follows the input linearly. The value of $R$ for other Networks varies between 0.56119 and 0.75112 shows poor regression. The learning time is almost equal for all the networks as the number of Epochs lies between 816 and 910 not showing any significant difference. Hence it is evident that the ELMAN network acting as a recurrent network with the tuned weights yields faster convergence with minimal MSE and less iterative process. As the recurrent layer is found to copy one step delay of the hidden layer and as recurrent link layer stores the detailed information of the hidden layer naturally shows that the ERNN suits best for the time-series prediction process.

5. Conclusion

In this paper the performance of different neural networks namely FFNN, CFFNN, ERNN and TDNN for the prediction of outlet temperature of a liquid-saturated steam heat exchanger are presented. The results prove that the Elman recurrent neural network achieves a better performance by reducing the mean square error and thereby increasing the prediction accuracy. The ERNN model will provide better predictions because it uses the additional information contained in the previous values using the feedback context layer.

6. References