CHAPTER 6

IMPLEMENTATION OF ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM

6.1 INTRODUCTION

Adaptive Neuro-Fuzzy Inference System (ANFIS) is a method of fuzzy inference employed in the structure of networks that are adaptive. The ANFIS maps inputs with the outputs based on the intelligence of humans and the specified input-output information pairs. The merits of ANFIS are its ability for rapid learning, nonlinear capability, and adaptation ability. In this chapter, the optimal cluster centers obtained using the subtractive clustering algorithm in Chapter 4 is fed as input to the ANFIS to analyze its performance in finding out the quantity of alloying elements that have to be added during the ladle refining process of steel making with reduced computational error to produce S235JRG2 steel grade.

6.2 ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM

ANFIS is an adaptive system (Jang 1993; Mohammad et al. 2010; Oyediran & Adeyemo 2013). It includes nodes which are connected through directional links. Furthermore, some or entire nodes are adaptive, which signifies that their outputs rely on the factors relating to these nodes. The learning rule denotes how these factors have to be varied to reduce the error function. Either backpropagation learning algorithm alone or a mixture of backpropagation learning algorithm and least squares method are used by
ANFIS to adjust the parameters of the Sugeno type fuzzy rule based system. The Figure 6.1(a) shows the first order Sugeno fuzzy model with two inputs and two rules and Figure 6.1(b) shows the ANFIS structural design (Jang et al. 1997). In Figure 6.1(b), the round nodes are the static nodes, while the nodes represented in square form are adaptive nodes.

\[ z_1 = g_1x + h_1y + s_1 \]

\[ z_2 = g_2x + h_2y + s_2 \]

\[ z = \frac{m_1z_1 + m_2z_2}{m_1 + m_2} = m_1z_1 + m_2z_2 \]

(a)

(b)

Figure 6.1  (a) First order Sugeno fuzzy model with two inputs and two rules (b) Corresponding ANFIS structural design
For ease, it is assumed that the considered fuzzy rule based system has \( x \) and \( y \) as two inputs and \( z \) as one output. The two fuzzy if-then rules meant for a first order Sugeno fuzzy model (Takagi & Sugeno 1983; Takagi & Sugeno 1985; Sugeno & Kang 1988) is as follows:

If \( x \) is \( U_1 \) and \( y \) is \( V_1 \) THEN \( z_1 = g_1x + h_1y + s_1 \)

If \( x \) is \( U_2 \) and \( y \) is \( V_2 \) THEN \( z_2 = g_2x + h_2y + s_2 \)

The \( i^{th} \) node output in the layer \( l \) is represented as \( T_{i,l} \). The ANFIS has five layers whose functions are explained as follows:

**Layer 1**

Each and every node \( i \) in layer one is an adaptive node with a node function,

\[
T_{1,i} = \mu_{U_{i}}(x) \quad \text{for} \quad i = 1, 2, \text{(or)}
\]

\[
T_{1,i} = \mu_{V_{i-2}}(y) \quad \text{for} \quad i = 3, 4,
\]

Here the input to the \( i^{th} \) node is represented as \( x \) (or \( y \)) and linguistic label related to this node is denoted as \( U_i \) (or \( V_{i-2} \)). The output \( T_{i,l} \) denotes the grade of membership of a fuzzy set \( U(= U_1, U_2, V_1 \text{ or } V_2) \). It also indicates the extent to which the specified \( x \) (or \( y \)) input term meets the expectations of the quantifier which is represented as \( U \). For \( U \), the membership function might be of any kind, but for illustration purposes the bell shaped function is used and is given by,

\[
\mu_{U_i}(x) = \frac{1}{1 + \left(\frac{x - c_i}{a_i}\right)^{2^b}} \quad \text{(6.2)}
\]
where $a_i, b_i, c_i$ are the premise parameters that have to be learnt.

**Layer 2**

Each node in this layer denoted as $\Pi$ are static. The output of these nodes is the multiplication of each signal that is arrived and is specified by,

$$T_{2,j} = m_i = \mu_{u_i}(x)\mu_{v_i}(y), \quad i = 1, 2$$  \hspace{1cm} (6.3)

In this layer the output of every node specifies the rules firing strength.

**Layer 3**

The nodes denoted as $\Sigma$ in this layer are static nodes which do the normalization of the rules firing strength.

$$T_{3,i} = \bar{m}_i = \frac{m_i}{m_1 + m_2}, \quad i = 1, 2$$  \hspace{1cm} (6.4)

**Layer 4**

In this layer each node is adaptive and its outputs are specified by,

$$T_{4,j} = \bar{m}_i z_i = \bar{m}_i (g_i x + h_i y + s_i)$$  \hspace{1cm} (6.5)

where $g_i, h_i, s_i$ are the consequent parameters and $\bar{m}_i$ represents the outputs from the third layer.

**Layer 5**

The static node in this layer denoted as $\sum$ is used for estimating the overall outcome as the sum of all the arriving signals.
The structure of ANFIS is not distinct. Some of the layers can be merged to produce the similar outcome. The role of learning algorithm for this structural design is to adjust each premise parameter \( a_i, b_i, c_i \) to generate the final result of ANFIS equivalent to the data that has been trained. In Figure 6.1(b) it is noted that, if the values of the premise parameters \( a_i, b_i, c_i \) are fixed, the net outcome might be represented as linear mixture of consequent parameters \( g_i, h_i, s_i \). Therefore, the final output represented as \( z \) in Figure 6.1(b) could be revised as,

\[
z = \frac{m_1}{m_1 + m_2} z_1 + \frac{m_2}{m_1 + m_2} z_2 \\
= \bar{m}_1 z_1 + \bar{m}_2 z_2 \\
= \bar{m}_1 (g_1 x + h_1 y + s_1) + \bar{m}_2 (g_2 x + h_2 y + s_2) \\
= (\bar{m}_1 x) g_1 + (\bar{m}_1 y) h_1 + (\bar{m}_1) s_1 + (\bar{m}_2 x) g_2 + (\bar{m}_2 y) h_2 + (\bar{m}_2) s_2 \quad (6.7)
\]

that is linear in the modifiable or consequent parameters denoted as \( g_1, h_1, s_1, g_2, h_2, \) and \( s_2 \). As a result, optimal values for the consequent parameters can be found effortlessly by merging the least-squares method with the gradient descent method, when the premise parameters \( a_i, b_i, c_i \) are fixed. If the premise parameters are not fixed then the learning algorithm will result in a slower convergence (Jang 1991).
6.3 IMPLEMENTATION AND RESULTS

The optimal cluster centers obtained using the subtractive clustering algorithm, given in Table A 2.1, are fed as training data to ANFIS and the results obtained for each training data output is shown in Figures 6.2-6.7. Since ANFIS is a Multiple Input and Single output (MISO) system, the steel dataset containing Multiple Input and Multiple Output (MIMO) is transferred to six MISO systems for implementation. The simulation is carried out using MATLAB R2014a. From Figures 6.2-6.7, it is observed that the ANFIS output is very much closer to the training data output.

![Figure 6.2 Performance of ANFIS for training data output Carbon (C in kg/Mg)](image-url)
Figure 6.3 Performance of ANFIS for training data output Silicon (Si in kg/Mg)

Figure 6.4 Performance of ANFIS for training data output Manganese (Mn in kg/Mg)
Figure 6.5  Performance of ANFIS for training data output Phosphorous (P in kg/Mg)

Figure 6.6  Performance of ANFIS for training data output Sulfur (S in kg/Mg)
Figure 6.7  Performance of ANFIS for training data output Aluminium (Al in kg/Mg)

The rule view for each output parameter of training data is shown in Figures 6.8-6.13. In the rule view shown in Figures 6.8-6.13, the amount of each output alloying elements can be determined for various values of the input parameters. The value of the input parameters can be altered by moving the red line indices as shown in Figures 6.8-6.13.
Figure 6.8  Rule view obtained using ANFIS for training data output Carbon (C in kg/Mg)

Figure 6.9  Rule view obtained using ANFIS for training data output Silicon (Si in kg/Mg)
Figure 6.10 Rule view obtained using ANFIS for training data output Manganese (Mn in kg/Mg)

Figure 6.11 Rule view obtained using ANFIS for training data output Phosphorous (P in kg/Mg)
Figure 6.12  Rule view obtained using ANFIS for training data output
Sulfur (S in kg/Mg)

Figure 6.13  Rule view obtained using ANFIS for training data output
Aluminium (Al in kg/Mg)
The RMSE obtained using ANFIS for each training data output is given in Table 6.1. The RMSE signifies the square root of the average square difference between the ANFIS outputs and the training data outputs. Lesser error value implies better performance, whereas zero represents no error. The Equation (6.8) is used to calculate the RMSE.

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (f_i - td_i)^2}
\]  

(6.8)

where \(f_i\) is the ANFIS outputs, \(td_i\) is the training data outputs and \(N\) is the number of training data. The training data output \(td_i\) denotes the output parameters of optimal cluster centers, which is given in Table A 2.1. From Table 6.1, it is perceived that the RMSE is optimal for each training data output and the average RMSE is found to be 5.89308e-07.

Table 6.1 RMSE obtained using ANFIS for each training data output

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Training Data Output</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Carbon (C)</td>
<td>4.1774e-07</td>
</tr>
<tr>
<td>2</td>
<td>Silicon (Si)</td>
<td>8.98233e-07</td>
</tr>
<tr>
<td>3</td>
<td>Manganese (Mn)</td>
<td>1.97352e-06</td>
</tr>
<tr>
<td>4</td>
<td>Phosphorous (P)</td>
<td>2.63327e-09</td>
</tr>
<tr>
<td>5</td>
<td>Sulfur (S)</td>
<td>2.99141e-10</td>
</tr>
<tr>
<td>6</td>
<td>Aluminum (Al)</td>
<td>2.43424e-07</td>
</tr>
</tbody>
</table>
6.4 PERFORMANCE COMPARISON OF THE PROPOSED INTELLIGENT TECHNIQUES

In Chapter 4, the performance of subtractive clustering algorithm was found to be better when compared with FCM clustering in terms of fuzzy rule optimization and RMSE, for computing the amount of alloying elements that have to be added at the time of ladle refining process in steel making. In Chapter 5, the BFGS Quasi Newton backpropagation algorithm showed better performance when compared with Levenberg-Marquardt backpropagation algorithm and Resilient backpropagation algorithm in terms of optimizing the error rate obtained using the subtractive clustering algorithm. Hence the performance of subtractive clustering algorithm and BFGS Quasi Newton backpropagation algorithm is compared with the performance of ANFIS evaluated in this chapter and is given in Table 6.2.

Table 6.2 Performance comparison of subtractive clustering algorithm, BFGS Quasi Newton backpropagation algorithm and ANFIS

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subtractive clustering algorithm</td>
<td>2.9719</td>
</tr>
<tr>
<td>BFGS Quasi Newton backpropagation algorithm</td>
<td>0.0254</td>
</tr>
<tr>
<td>ANFIS</td>
<td>5.89308e-07</td>
</tr>
</tbody>
</table>

From Table 6.2, it is evident that merging of fuzzy clustering method with ANFIS has proved its supremacy in determining the amount of alloying elements that have to be added during the process of ladle refining to produce S235JRG2 steel grade with reduced computational error.
6.5 SUMMARY

This chapter has presented the description of ANFIS technique and evaluated its performance by applying it to the optimal cluster centers obtained using the subtractive clustering algorithm. The performance of ANFIS is analyzed by comparing it with the performance of the subtractive clustering algorithm, and BFGS Quasi Newton backpropagation algorithm in terms of computational error. Based on the analysis performed, it is observed that merging fuzzy clustering method with ANFIS has resulted with a reduced RMSE of 5.89308e-07 in determining the amount of alloying elements that have to be added during the process of ladle refining in steel making to produce S235JRG2 steel grade.