Chapter 5
CHAPTER 5

AN ADAPTIVE FLC USING EXPERT SYSTEM BASED LEARNING APPROACH

5.1 INTRODUCTION

Fuzzy Logic theory is used to implement human thinking and decision making process in the computer (Scharf E.M. 1985). The FLC discussed and implemented in the previous chapters has static rule base which is considered to be the brain of FLC.

The number of fuzzy sets defined in the input and output universe of discourse and the number of fuzzy rules in the rule base heavily influence the complexity of fuzzy system where complexity includes the computational complexity and space complexity.

Human experts are constantly updating and upgrading their expertise. A lot of expertise improvement comes as a by-product by acting, the role of the expert, by confronting new and difficult situations. The expert studies the situation, thinks of a good solution or takes advice from other experts and stores the information in his memory and thereby knowledge is enhanced. In the similar way rule base of the FLC is updated by using an expert system which updates the original rule base into a current rule base with less number of rules by statistical or optimal method. In the mean time the quality of the control system is evaluated by minimum error performance criterion. The original rule base is switched i.e. all the rules are evoked, when the system deteriorates due to change in set-point or process condition.

With the objective of retaining an effective man machine interface and keeping the rule base computation optimal (as contrasted to computation-intensive adaptive control algorithms), use of expert system based paradigm for FLC is presented in this chapter. The software for the FLC algorithm has been written in C++. The proposed
FLC is applied to a simulated simple second order system and the performance is shown in graphs.

The key issue in the design of expert control systems for industrial processes is the acquisition of process/plant knowledge (Scharf E.M., Mondie N.J. and Mamdani E.H. 1983). To this end, two distinct approaches have emerged (Sudkamp T. and Hammell R.J. II, 1994) viz. (i) to capture the expert knowledge (heuristics in the rule based format (Tong R.M. 1977)) of the process personnel (process engineers, process operators, etc.) (ii) to establish casual accounts of the physical mechanisms without invoking the mathematics of continuously varying quantities i.e. to deduce a qualitative model of the process (Maiers J. 1985). Both these approaches have been applied in pervious applications discussed in chapter 3 and 4.

In this study the term adaptive nature of FLC is in the sense, optimization of the number of rules in the rule base (Gunnar Johannsen 1991). The expert system used in this application is a learning or knowledge acquisition system, rather than a problem solving tool (Jesperson T. 1981).

The design procedure of rule base of FLC is discussed in section 2. The idea of expert system based adaptive learning is given in section 3. The proposed controller is implemented to a simulated system which is discussed in section 4 and the result and discussion are given in section 5.

5.2 FUZZY LOGIC CONTROLLER

The Fuzzy Logic Controller (FLC) consist of three modules namely Fuzzification Module, Rule base, Defuzzification Module. In this study the FLC Type II is used for the proposed improvement and simulation. The fuzzification module used in chapter 2 section 2.2.2 is used. The rule base consists of 64 linguistic rules which are enumerated in Table 2.3. The output of the rule base should be converted into crisp value. This task is done by defuzzification module Maxima criterion, Mean of Maxima
and Centre of Area method of Defuzzification are discussed in chapter 2. Center of Area method of defuzzification is used here, since the study has been done for implementation in ATG system discussed in previous chapter.

5.2.1 Design of rule base for a static FLC

Rule base consists of a set of rules in the form of fuzzy logic operations, i.e. IF-THEN rules. These rules are programmed in the computer to form the rule base. Based on the type of application and resources of the project, either one or more combinations of the methods discussed in chapter 2, can be used for design of rule base. In this study rule base is developed using Mac Vicar-whelan tables.

5.2.2 Mac Vicar-Whelan method of Rule Base

Consider a discrete set-point controller, the error $e(k)$ and change in error $\Delta e(k)$ are the input to the controller where error $e(k)$ at time instant $k$ is the difference between process output and the set point value. The output of PI controller $u(k)$ may be given by (5.1), which drives or manipulate the process

$$u(k) = u(k-1) + \Delta u(k) \quad \ldots \ldots \ (5.1)$$

The value of $\Delta u(k)$, the incremental change in controller output is determined by $e(k)$ and $\Delta e(k)$. The rule base consists of rules of the form "IF $e(k)$ is POSITIVE BIG and $\Delta e(k)$ is NEGATIVE SMALL THEN make $\Delta u(k)$ NEGATIVE MEDIUM" where positive big, negative small and negative medium are linguistic variables, represented by fuzzy sub-set of the universe of discourse of the controller variables.

The input to the Fuzzy Logic Controller $e(k)$, $\Delta e(k)$ and output of FLC viz. $\Delta u(k)$ are fuzzified into 8 fuzzy sub-sets respectively. Using the following meta-rules Mac Vicar-Whelan designed the rule base of FLC for simple system in a table form (Sembi B S. 1980)
1 If both \( e(k) \) and \( \Delta e(k) \) are zero, then maintain the present control action i.e. 
\( \Delta u(k) = 0 \)

2 If conditions are such that \( e(k) \) will go zero at a satisfactory rate considering 
\( \Delta e(k) \), then maintain the present control action i.e. \( \Delta u(k) = 0 \)

3 If \( e(k) \) is not self correcting, then control action \( \Delta u(k) \) should be changed 
according to the sign and magnitude of \( e(k) \) and \( \Delta e(k) \)

Using the Mac. Vicar-Whelan's Table Fig. 2.4, the possible rules that can be 
formed is taken as the rule base. The rule base consists of 64 rules which are given in 
Table 2.3

5.2.3 Drawback of Static Rule Base

In general a rule in the rule base is of the form

\[
\text{WHEN (} U_1 \text{ is } A_{i1}) \text{ AND (} U_2 \text{ is } A_{i2}) \text{ AND} \ldots \text{ (} U_r \text{ is } A_{in}) \text{ THEN (} V \text{ is } D_i) 
\]

Where \( U_1, U_2 \ldots U_r \) are input variables and \( V \) is the output variable (In case of 
this problem \( r = 2 \) and \( U_1 \) is error \( e \) and \( U_2 \) is change in error \( \Delta e \) and \( V \) is change in 
controller output \( \Delta u \))

\( X_i \) is the universe of discourse of \( U_i \) and \( A_{ii} \) is a fuzzy sub set of \( X_i \) and \( Y \) is the 
universe of discourse of \( V \) and \( D_i \) is the fuzzy subset of \( Y \).

Assume the input to FLC, \( U_i = X_i \), the procedure for reasoning used in the 
Mandani - Zadeh method consists of the following steps (Yamazaki T. 1982):

1 Calculate the firing level \( \tau \) of each rule, \( \tau_i = \text{Min} |A_{ii}(X_i)| \).
2 Calculate the output of each rule as a fuzzy subset \( F_i \) of \( Y \) where 
\[
F_i(y) = \text{Min} \left[ \tau_i, D_i(y) \right]
\]

3 Aggregate the individual rule output to get a fuzzy subset \( F \) of \( Y \) where 
\[
F(y) = \text{Max} \left[ F_i(y) \right]
\]

The computation time of rule base with large number rules is more than that with few number of rules. Thus the computational efficiency decreases when the number of rules increases (Xian-Tu Peng 1988). Out of the 64 rules in the static rule base, for a particular operation or for a particular set-point variation only 40 rules are used or fired. The other rules are fired only when the process is pushed to extreme level of stability. Even though FLC is said to be adaptive, as long as it has a static rule base, FLC is a deterministic controller and does not have intelligence.

5.3 EXPERT SYSTEM BASED LEARNING APPROACH

As discussed in the previous section, the FLC has static rule base which consists of 64 rules using Mac Vicar-Whelan table. The membership function of linguistic variables are bijective transformation type and Center of Area method of defuzzification strategy is followed. The rule base consists of redundant rules to take care of situation arising due to extreme change in load or set point. The rules taking care of extreme conditions may not be used at all. But still they have to be executed.

Experts operate some plants which are more complicated to be automated using conventional methods. When they encounter simple situation they use few thumb rules and when the situation is more complicated they use more thumb rules. Similarly if there is a mechanism which eliminates the rules that are not used, (when the plant is running smoothly) the system may be said to have some intelligence.

For example an naive auto mechanic does diagnostic procedure to identify the fault, when he has to repair an automobile. If he is experienced, he uses this expertise
to identify the fault immediately without doing the diagnostics. He goes through the log book and from the frequency of the types of previous faults, he identifies the fault quickly. If he finds more occurrence of the fuel-related problem, he investigates in fuel supply system. Otherwise, he investigates electrical systems. Thus, he is able to identify the problem quicker than the heuristic method. The same idea may be used in FLC also and unnecessary wastage of computational time in computing all rules can be avoided.

5.3.1 Expert System Based Adaptive Rule Base

A simple data base is created in which the firing level of all the rules enumerated in the current rule base is stored. The data base has the firing level of each rule for the past N instants. The rule which has probability for firing level in the range of $[0, 0.2]$ is more than 0.5, the same will be removed from the rule base at the N+1\textsuperscript{th} instant. In this manner, rules which are having low firing level for the particular plant operation is removed. But when the system becomes unstable or the system reaches extreme condition all the rules will be reinstated to ensure stability and smooth controlling of process. A fresh data base will be initiated with the current rule base comprising of all the rules.

5.3.2 Algorithm

The algorithm for the expert system based adaptive rule base is given below. The flow chart is shown in Fig. 5.1.

(1) Initiate the current rule base with all possible rules and fill up the data base with firing level 1 for the N previous instants

(2) Statistical calculation, determination of probability and updating of the current rule base
(3) System performance test and if the system's performance is bad continue from step 1 else

(4) update the data base and continue from step 2.

The data base is shown in Fig. 5.2. The standard deviation \( \sigma \) of firing value for each rule is calculated using the following equation (Gupta S.P. 1990)

\[
\sigma = \sqrt{\frac{(X - \bar{X})^2}{N}} \quad \text{......... (5.2)}
\]

where \( \bar{X} \) is the mean and \( N \) is the number of data in the data base.

Using normal distribution probability methods the probability for the firing level of a particular rule to be in \([x_0, x_1]\) is calculated using the following formula

\[
p = \frac{1}{\sigma \sqrt{2\pi}} \int_{x_0}^{x_1} \exp\left(-\frac{(X - \bar{X})^2}{2\sigma^2}\right) dX \quad \text{......... (5.3)}
\]

If the probability for a particular rule to have a firing level in the range \([0, 0.2]\) is more than 0.5 at any instant the rule is removed from the rule base. Then system performance test is done.

The error and change in error are important indicators of the goodness of the controller in operation. If the following fuzzy logic conditions are satisfied

1. \( e(t) \) PZE and \( \Delta e(t) \) NZE
2. \( e(t) \) NZE and \( \Delta e(t) \) PZE
3. \( e(t) \) PS and \( \Delta e(t) \) NS
4. \( e(t) \) NS and \( \Delta e(t) \) PS
then the performance of the controller is good (qualitative term). The system is tested if one of the conditions given above is satisfied. If it satisfies, the performance is well if not, the performance of the controller need improvement. Thus system performance test is done.

5.4 SIMULATION STUDY

This simulation study is mainly undertaken for implementing the proposed paradigm for ATG system. So a mathematical model similar to the ATG system is used i.e. a second order over damped system. The transfer function of the same is given in (5.4). With sampling time 1 second the equation (5.4) is discretized using Z-Transformation technique and software developed using C programming language (Louis Baker 1989).

\[
G(S) = \frac{c(S)}{r(S)} = \frac{K}{(\tau_1 S + 1)(\tau_2 S + 1)} \quad \ldots \ldots (5.4)
\]

Where \(K = 1\), \(\tau_1 = 20\) sec, \(\tau_2 = 10\) sec.

The set-point to the system is sinusoidal as in the case of ATG system. The optimal value of \(N\) i.e. the number of instants to be stored in the data base should be arrived. So data base with 50, 100, 200, 300, 400 instants data are designed and tested.

The probability for each rule to fire in the range of \([0,0.2]\) is calculated with above mentioned data base with varying values of \(N\) and shown as bar charts in Fig. 5.3-5.7. From this observation it is clear that the probability values do not change much for data above 100 instants (for the sinusoidal type of set-point given).
As proposed in section 5.3 an adaptive rule base is designed with a data base having data of previous 100 instants. The set-point tracking performance of the expert system based FLC is shown in Fig. 5.8.

5.5 RESULT AND DISCUSSION

After the current rule base is stabilized, the number of rules used is only 35. Even then the performance of the controller is good. The saving in computation time is observed and the same is approximately 25 milli second.

The advantage of the proposed expert system based paradigm is that the rule base can be formed with more number of rules acquired from the process operators and they can utilize their experiential knowledge to incrementally extend the rule base and thus enhance controller performance. The disadvantage with varying number of rules as proposed is the variable time between samples until the controller has stabilized its rule base. The drawbacks with data base proposed for expert system is that they are generally sparse and they require large storage space.
Fig. 5.1 Flow chart for expert system based adaptive rule base
<table>
<thead>
<tr>
<th>Instant</th>
<th>Rule 1</th>
<th>Rule 2</th>
<th>Rule 3</th>
<th>Rule 64</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a_{11}</td>
<td>a_{21}</td>
<td>a_{31}</td>
<td>a_{N1}</td>
</tr>
<tr>
<td>2</td>
<td>a_{12}</td>
<td>a_{22}</td>
<td>a_{32}</td>
<td>a_{N2}</td>
</tr>
<tr>
<td>3</td>
<td>a_{13}</td>
<td>a_{23}</td>
<td>a_{33}</td>
<td>a_{N3}</td>
</tr>
<tr>
<td>4</td>
<td>a_{14}</td>
<td>a_{24}</td>
<td>a_{34}</td>
<td>a_{N4}</td>
</tr>
<tr>
<td>\infty</td>
<td>a_{1\infty}</td>
<td>a_{2\infty}</td>
<td>a_{3\infty}</td>
<td>a_{N\infty}</td>
</tr>
</tbody>
</table>

Fig 5.2 Data-base
Fig 5.3  Bar chart for probability of rules to fire in [0, 0.2] using data base of 50 instants
Fig 5.4 Bar chart for probability of rules to fire in [0, 0.2] using data base of 100 instants
Fig. 5.5  Bar chart for probability of rules to fire in [0, 0.2] using data base of 200 instants
Fig. 5.6  Bar chart for probability of rules to fire in [0, 0.2] using data base of 300 instants
Fig. 5.7 Bar chart for probability of rules to fire in [0, 0.2] using data base of 400 instants.
Performance of expert system based adaptive FLC implemented to second order overdamped system.