A Comparative Analysis Between Developed Integral Controller and Fuzzy Logic Based Controller for Automatic Generation Control

Kalyan Chatterjee         T.Ghose              B.M.Karan

In this paper, design and implementation of integral and Fuzzy Logic based controller for automatic generation control are investigated and tested for a particular power system network[1]-[2]. For integral controller, a very simple but a fast search technique named as Hooke-Jeeve’s method [4] optimizes the integral constant value (K_i). Fuzzy logic based controller for the same purpose is also developed and the comparative performance of these two methods is given.

II. DESIGN OF CONVENTIONAL CONTROLLER.

A. Performance Index:

As we know the design of a control system is an attempt to meet a set of specifications, which define the overall performance of the system in terms of certain measurable quantities. A number of performance measures have been introduced so far in respect of dynamic response to step input (M_p, t_r, t_p, t_s) and steady state error e_s to both step and higher order inputs. A single performance index could be established on the basis of which one might describe the goodness of the system response, then the design procedure will become logical and straightforward.

In adaptive control systems we require a performance index, which is a function of the variable system parameters. Extremum (minimum or maximum) value of this index then corresponds to the optimum set of parameter values. Other desirable features of a performance index are its sensitivity, i.e. its power to clearly distinguish between an optimum and non-optimum value.

A number of performance indices are used in practice, the most common being the Integral of the Squared Errors (ISE), given by

\[ ISE = \int_0^\infty e(t)^2 \, dt \]

Here \( e(t) \) is the error at time \( t \). The error in the present problem is the frequency error. The performance indices (PI) or the objective function for this study are of the form:

\[ ISE = \int_0^\infty \alpha (\Delta f)^2 \, dt \]

Minimization of ISE by adjusting system parameters is a good compromise between reduction of rise time to limit the effect of large initial error, reduction of peak overshoot and reduction of settling time to limit the effect of small error lasting for a long time. To get the optimum value of the integral constant, a simple search technique called Hooke-Jeeve’s algorithm is used.
B. Development of the Solution by Hooke-Jeeve’s Method

The algorithm of Hooke method is a direct search method. Only the functional values at different points are used to constitute the search. In direct search method each variable can be modified either in positive or in negative direction. The Hooke-Jeeve's method [4] comprises of an iterative application of an exploratory move in the locality of current point and a subsequent jump using the pattern move.

In the exploratory move the current point is perturbed in positive and negative directions along each variable one at a time and the best point is recorded. The current point is changed to the best point at the end of each variable perturbation. If the point found at the end of all variable perturbations is different than the original point, the exploratory move is a success, otherwise the exploratory move is a failure. In any case, the best point is considered to be the outcome of the exploratory move.

In pattern move the final point \( x^{(k+1)} \) is found by jumping from the current best point \( x^{k} \) along a direction connecting the previous best point \( x^{(k-1)} \) and the current base point \( x^{k} \) as follows:

\[
x^{(k+1)} = x^{(k)} + (x^{(k)} - x^{(k-1)})
\]

If the pattern moves do not take the solution to a better region, the exploratory move is not accepted and the extent of the exploratory search is reduced. An automatic load frequency control for single area power system is simulated with the help of Matlab Simulation toolbox. Software for Hooke-Jeeve’s method is developed in Matlab Programming to find out the value of optimum \( K_i \).

III. FUZZY LOGIC CONTROL STRUCTURE:

In conventional control, the amount of control is determined in relation to a number of data inputs using a set of equations to express the entire control process. Expressing human experience in the form of a mathematical formula is a very difficult task, if not an impossible one. Fuzzy logic provides a simple tool to interpret this experience into reality. Fuzzy Logic controller(FLC) can be simply represented in four parts as shown in Fig.1.

- **Fuzzificatation module**: the function of which are, first, to read, measure, and scale the control variable (e.g. frequency error) and second, to transform the measured numerical values to the corresponding linguistic (fuzzy) variables with appropriate membership values.

- **Knowledge base**, which includes the definitions of the fuzzy membership functions for each control variable and the necessary rules that specify the control goals using linguistic variables.

- **Inference mechanism**, which is the kernel of the FLC. It should be capable of simulating human decision making and influencing the control actions based on fuzzy logic.

- **Defuzzification module**, which converts the inferred decision from the linguistic variables back to numerical values.

IV. DESIGN OF FUZZY LOGIC BASED AGC CONTROLLER

The design process of a fuzzy logic based AGC controller may be split into the following steps described below:

A. Selection of control variables

As mentioned above, the first step is to select the controller input and output variables. For single area system the frequency error (\( \Delta f \)) and the change in frequency error (\( \Delta f/\Delta t \)) are taken as inputs and the output is the controlled variable (\( \Delta P_c \)) as shown in Fig.-2.

B. Membership function Definition:

The next step in fuzzification module is to transform the measured errors (\( \Delta f \) and \( \Delta f/\Delta t \)) and output(\( \Delta P_c \)) to the corresponding linguistic variable with appropriate membership values. Each control is quantized overlapping of linguistic variables or fuzzy sets. A set of membership functions defined for seven linguistic variables NB, NM, NS, Z, PS, PM and PB, which stand for Negative Big, Negative Medium, Negative small, Zero, Positive small, Positive Medium and Positive Big respectively shown in Fig. 3.

![Fig.1: Schematic diagram of the FLC building blocks](image)

![Fig.2: Block Diagram For Isolated area usingController](image)

![Fig.3: Membership Functions Of Seven Linguistic Variables](image)
C. Rule Creation and inference:
The next task is to create Fuzzy rules using linguistic variables to get an output set. In this case the Fuzzy controller of two Fuzzy variables and one output Fuzzy variable, each quantized to seven Fuzzy sets. This leads to a $7 \times 7$ rule matrix shown in Table-I.

<table>
<thead>
<tr>
<th>Frequency Error</th>
<th>NB</th>
<th>NM</th>
<th>NS</th>
<th>Z</th>
<th>PS</th>
<th>PM</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NB</td>
<td>FE</td>
<td>FB</td>
<td>PM</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
</tr>
<tr>
<td></td>
<td>NM</td>
<td>FE</td>
<td>FM</td>
<td>PS</td>
<td>Z</td>
<td>Z</td>
<td>ZS</td>
</tr>
<tr>
<td></td>
<td>NS</td>
<td>FM</td>
<td>FM</td>
<td>PS</td>
<td>Z</td>
<td>ZS</td>
<td>NS</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>FM</td>
<td>FS</td>
<td>PS</td>
<td>Z</td>
<td>ZS</td>
<td>NS</td>
</tr>
<tr>
<td></td>
<td>PS</td>
<td>FS</td>
<td>FS</td>
<td>Z</td>
<td>Z</td>
<td>ZS</td>
<td>NS</td>
</tr>
<tr>
<td></td>
<td>PM</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>ZS</td>
<td>NS</td>
</tr>
<tr>
<td></td>
<td>PB</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>ZS</td>
<td>HB</td>
</tr>
</tbody>
</table>

TABLE-I

D. Defuzzification Strategy.
Defuzzification is a process of converting the FLC inferred control actions from fuzzy values to crisp values. This process depends on the output Fuzzy set, which is generated from the fired rules. In this case, Centroid method uses for Defuzzification.

V. Simulation and Results
The proposed controllers are applied to a single-area non-reheat thermal system [1]–[2]. For conventional controller compute the optimum parameter values. A unit step load Fig.4. Simulation Model For Single-Area Using

Integral Controller
change (.01P.U) is assumed and the performance index is minimized using a Hooke-Jeeve’s algorithm. To compute the performance index a digital simulation was performed in MATLAB simulink as shown in Fig- 4.

Over a simulation time period of 15 sec., for each of the individual parameter $K_i$. Table II shows that optimum $K_i$ value is found 0.9625 after 6th iteration for 0.01 P.U load change and change of $\Delta f$ with time shown in Fig.5.

For the same problem by using Fuzzy Logic Controller the change in frequency with time shown in Fig.6. Fuzzy Logic simulation model, has been developed with the help of Fuzzy Logic Toolbox.

Table III shows the comparative analysis in respect of overshoot and settling time. The results show that Fuzzy Logic Controller has better dynamic response.

![Fig.5: Comparison of Integral and Fuzzy Controller](image)

VI. Conclusions
In this paper Fuzzy Logic And Integral Controllers are utilized for Automatic Generation control of single area system and compared to the results with the conventional controller. From the results in case of single area system was reduced from -0.058 Hz to -0.025 Hz in comparison with the integral controller and settling time is also reduced drastically from 11 seconds to 4 seconds. It is very clear that Fuzzy Logic Controller is giving good performance due to dynamic behavior.

<table>
<thead>
<tr>
<th>Overshoot</th>
<th>Settling time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy Controller</td>
<td>Integral Controller</td>
</tr>
<tr>
<td>-0.025 Hz</td>
<td>-0.058 Hz</td>
</tr>
<tr>
<td>4 sec</td>
<td>11 sec</td>
</tr>
</tbody>
</table>

VII. References