Performance Analysis of Tuned PID Controller for Load Frequency Control using Optimization Technique

Kiran Kushwaha, Ram Kumar Denjare

Research scholar (M.Tech), Assistant Professor
Department of Electrical Engineering, Kalinga University, New Raipur, Chhattisgarh

Abstract - This paper presents a practical model for optimization of the load frequency control of a two area system. The proposed model is framed with the thermal generating units and it is a non-linear power system. For design and analysis of the proposed method the Proportional Integral Derivative (PID) controller is used. A unique objective function is formulated which considering the transient specifications. A Particle Swarm Optimization (PSO) algorithm and Simulated Annealing (SA) is used to obtain the best control parameters. The results and characteristics are compared and find which proposed model gives better performances.

Index Terms - two area power system; load frequency control; PID controller; particle swarm optimization, Simulated Annealing.

I. Introduction

The main objective of the controller is to quickly reach the target to a new set point so that the stability of the system is maintained quickly. For that purpose various tuning techniques are developed. PID controllers are mostly used now a days as these can be easily implemented and its performance is quite better. Many Intelligence techniques are being introduced in the controller so as to enhance the system characteristics performance. By using these techniques the stability of the system is improved to a greater extent. In this paper different optimization techniques are applied for the best selection of PID controller parameters. Optimum tuning PID has been applied by many random search methods which includes particle swarm optimization algorithm (PSO)[9] and simulated annealing (SA). This paper focuses on optimal tuning of PID controller for the simple transfer function of LFC system using different adaptive search techniques. In this paper the best value of PID controller parameters is obtained by different tuning techniques applied through the algorithm, this parameter helps in generating the best selection of gains of the controllers. After obtaining the values these are being sent to the workspace which is then shared by the simulink model for the simulation process.

II. Load Frequency Control

Load frequency problems:

If the system is connected to a number of different loads in a power system then the system frequency and speed change with the governor characteristics as the load changes. If it is not required to keep the frequency constant in a system then the operator is not required to change the setting of the generator. But if constant frequency is required the operator can adjust the speed of the turbine by changing the governor characteristic as and when required.

Load Frequency Control:

Load Frequency Control (LFC) scheme basically incorporates an appropriate control system or an interconnected power system, which is having the capability to bring the frequencies of each area and the tie line powers back to original set point values or very nearer to set point values effectively after the sudden load change. The main objective of Load Frequency Control is to maintain the output power within a particular area with respect to change in system frequency. This can be achieved by the use of controllers.[1]

III. Tuning of PI controller

The most important controllers used now a days are the PID controllers. These are used in many industrial application. The PID controller has three control parameters namely proportional, integral and derivative controllers. The proportional part is due to the desired set point and is proportional to the current error value. Whereas the integral part and the derivative part accounts for the past errors.

A typical structure of a PID control system is shown in Fig. 1
The error signal is used to generate the proportional, integral, and derivative actions. The output of the PID controller then goes to the plant model which then gives the desired result. The mathematical expression of PID control is expressed in equation (1).

\[
\text{where } e(t) \text{ is the input signal to the plant model, is the error signal, which is given as } \text{ and } r(t) \text{ is the reference input signal.}
\]

When we increase the value of \( k_p \) then an increasing value of peak overshoot is obtained and decrease in steady state error occurs. That means as we increase the value of \( k_p \) more and more the system becomes unstable due to the increase in overshoot. On the other hand when \( k_p \) increases, the overshoot becomes smaller, and the speed of response becomes slower and when we increase the value of \( k_p \) the overshoot is very small with a slow rise time but the settling time is similar.

Mostly in practical purpose the pure derivative action is never used as it produces a derivative kick due to which an undesirable noise is also produced. This drawback can be overcome by replacing it with a first order low pass filter. Hence the laplace transform of the PID controller is represented as

A step input is then applied to the control system and the error performance criteria is being applied for reducing the error. Here in this paper we are using Integral Time Absolute Error(ITAE) criteria for reducing the error[7].

**IV. Intelligent Optimization Techniques**

**Particle Swarm Optimization (PSO)**

Particle swarm optimization (PSO) is an optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995. This is population based optimization technique which was inspired by the social behaviour of fish schooling and bird flocking.

The basic algorithm of PSO is[9,11]

**Step1** - At first the minimum and maximum value of the three controller parameters are being specified. This is done by selecting the population of individual which includes the searching point, its individual best value \( p_{\text{best}} \) and its global best value \( g_{\text{best}} \).

**Step2** - After that the fitness value is being calculated for each individual using the evaluation function.

**Step3** - Comparison of each individual is being done which is known as \( p_{\text{best}} \). The best value from \( p_{\text{best}} \) is denoted as is \( g_{\text{best}} \)

**Step 4** - After that the member velocity is being modified for each individual \( k \). 

\[
\text{where } v_i(t) \text{ is known value. When } g \text{ is } 1 \text{ then it represents the change in velocity of controller parameter } \. \text{When } g \text{ is } 2 \text{, then it indicates the change in parameter } k_i \text{. Similarly when } g \text{ is } 3 \text{ then it denotes the change in parameter } k_d.
\]

**Step5** - If \( \text{, then} \)

**Step6** - Modified the member of each individual .

Where and represent the minimum and maximum, respectively, of member of the individual. When \( k_p \) parameter indicates lower and upper bound which is indicated by and respectively. When \( k_i \) controller decides the which are indicated by and respectively. When \( k_d \) controller indicates the lower and upper bounds which are being indicated by and respectively.

**Step 7** - If the maximum value is reached through number of iteration then proceed to Step 8. or else proceed to Step 2.

**Step 8** - The latest individual which is now generated becomes the optimal controller parameter.

The Fig 4 shows that the flowchart of parameter optimizing procedure using PSO.
Simulated annealing

This optimization technique involves perturbation of the design variables and then observing the change in the objective function. If the solution is better than the current solution, then only the design variables are updated and the new solution is accepted according to the Metropolis algorithm based on Boltzmann probability [17]. The perturbations keep on reducing according to some reduction constant. The algorithm ends when the desired solution to the objective function is reached or the perturbations are too small for significant change in the objective function [22].

The basic algorithm of simulated annealing is,

Step1- First, generate a random solution.
Step2- After that calculate its cost using some cost function you’ve defined.
Step3- Now generate a random neighboring solution.
Step4- After that calculate the new solution’s cost
Step5- Compare them:
  - If \( c_{\text{new}} < c_{\text{old}} \): move to the new solution
  - If \( c_{\text{new}} > c_{\text{old}} \): maybe move to the new solution

Step6- Repeat steps 3-5 above until an acceptable solution is found or you reach some maximum number of iterations.

A flow chart of the algorithm is shown in Fig. 2.

---

**Fig. 4.** Flow chart for simulation of PSO based PI controller.

**Fig. 2** Flow chart for simulation of SA based PID controller
V. Result and Discussion

This paper discusses about the implementation of different adaptive controller tuning technique for the Load Frequency Control (LFC) model.

Case 1 Load Frequency Control using IAE criteria

Comparison of SA and PSO Using Integral Absolute Error (IAE)

At first the tuning technique is applied using IAE criteria for SA due to which the terminal voltage step response is obtained. Similarly PSO is applied to the for obtaining the response of the system. After obtaining the response of all the two methods the terminal voltage response is compared which is indicated in fig 5.

![Fig 5. Performance parameter of SA and PSO for IAE](image)

Table I gives the review of different comparison values when Integral Absolute Error minimization is done. Considering the rise time response for SA is quite good and in case if the settling time and peak overshoot is considered then PSO gives much better response

Case 2 Load Frequency Control using ITAE criteria

Comparison of GA, SA and PSO Using Integral Time Absolute Error (ITAE)

The second error minimization technique named ITAE is then applied to the optimization method SA and PSO through which the magnitude of the step response is obtained accordingly. And the result of the three optimization methods are compared which are shown in fig 6 which shows that the response characteristics of PSO is quite better.
Table II

<table>
<thead>
<tr>
<th>Tuning method</th>
<th>kp</th>
<th>Ki</th>
<th>Kd</th>
<th>Mp%</th>
<th>Ts(s)</th>
<th>Tr(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>4.9275</td>
<td>0.5772</td>
<td>1.5066</td>
<td>54.4289</td>
<td>108.3747</td>
<td>4.6178</td>
</tr>
<tr>
<td>PSO</td>
<td>0.1000</td>
<td>3.5603</td>
<td>3.0116</td>
<td>37.4189</td>
<td>284.8151</td>
<td>4.4655</td>
</tr>
</tbody>
</table>

Comparison of SA and PSO for ITAE

Table II indicates different performance parameters when integrated time absolute error minimization technique is applied to the three intelligent optimization method. So in this case settling time is very less for SA as compared to the other methods. In case of peak overshoot and rise time PSO gives better parameters than SA.

Case 3 Load Frequency Control using ISE criteria

Comparison of SA and PSO Using Integral Square Error (ISE)

The comparison of the terminal voltage for the three optimization methods are now done using ISE criteria the response of which is shown in Fig 7 which purely indicates that the performance parameters for GA and SA is same.

Table III

<table>
<thead>
<tr>
<th>Tuning method</th>
<th>kp</th>
<th>Ki</th>
<th>Kd</th>
<th>Mp%</th>
<th>Ts(s)</th>
<th>Tr(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>0.1473</td>
<td>4.8793</td>
<td>0.5284</td>
<td>23.0817</td>
<td>291.7545</td>
<td>95.9085</td>
</tr>
<tr>
<td>PSO</td>
<td>1.8723</td>
<td>0.1000</td>
<td>0.2297</td>
<td>25.8965</td>
<td>137.7638</td>
<td>10.3790</td>
</tr>
</tbody>
</table>
From Table III it is clear that in Integral Square Error minimization technique if peak overshoot is considered then SA proves much better than PSO and if settling time and rise time is considered then PSO is much better than SA.

Case 4 Load Frequency Control using ITSE criteria

Comparison of SA and PSO Using Integral Absolute Error (ITSE)

At last the ITSE criteria is being applied to the load frequency model for all the three optimization techniques and their results are being compared. Fig 8 shows the response and comparison graph. The various controller and performance values are shown in Table IV.

![Fig 8 Performance parameter of SA and PSO for ITSE](image)

<table>
<thead>
<tr>
<th>Table IV</th>
<th>Comparison of SA and PSO for ITSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tuning method</td>
<td>kp</td>
</tr>
<tr>
<td>SA</td>
<td>4.9939</td>
</tr>
<tr>
<td>PSO</td>
<td>2.0676</td>
</tr>
</tbody>
</table>

After the comparison of SA and PSO for Integrated Time Square Error the various values of controller parameters are obtained. And from the above comparison it is seen that the settling time and rise time of SA is very large whereas for PSO it is very less. The peak overshoot for SA gives a good response.

VI Conclusion

In this paper various tuning methods are classified. According to that they are implemented for the stability enhancement of the system. Out of the various controller tuning at first the trial and error method was applied for the selection of parameters. But through result we come to know that the trial and error method was not appropriate. So we use the various time integral performance criteria. Various time integral performance criteria has been explained and their simulation results are compared. We apply the simulated annealing and particle swarm optimization technique to tune the PID parameters kp, ki and kd to meet the system performance criteria for various time integral performance criteria.

References


[27] A. Soundararajan, Member, IAENG, Dr. S. Sumathi, C. Sundar “Particle Swarm Optimization Based LFC and AVR of Autonomous Power Generating System” IAENG International Journal of Computer Science, 37:1, IJCS_37_1_10.


