ADAPTIVE FUZZY CONTROLLER TO CONTROL TURBINE SPEED

K. Gowrishankar, Vasanth Elancheralathan
Rajiv Gandhi College Of Engg. & tech., Puducherry, India
gowri200@yahoo.com, vasanth.elan@yahoo.com

Abstract: It is known that PID controller is employed in every facet of industrial automation. The application of PID controller span from small industry to high technology industry. In this paper, it is proposed that the controller be tuned using Adaptive fuzzy controller. Adaptive fuzzy controller is a stochastic global search method that emulates the process of natural evolution. Adaptive fuzzy controller have been shown to be capable of locating high performance areas in complex domains without experiencing the difficulties associated with high dimensionality or false optima as may occur with gradient decent techniques. Using Fuzzy controller to perform the tuning of the controller will result in the optimum controller being evaluated for the system every time. For this study, the model selected is of turbine speed control system. The reason for this is that this model is often encountered in refineries in a form of steam turbine that uses hydraulic governor to control the speed of the turbine. The PID controller of the model will be designed using the classical method and the results analyzed. The same model will be redesigned using the AFC method. The results of both designs will be compared, analyzed and conclusion will be drawn out of the simulation made.

Keywords: Tuning PID Controller, ZN Method, Adaptive fuzzy controller.

1 INTRODUCTION

Since many industrial processes are of a complex nature, it is difficult to develop a closed loop control model for this high level process. Also the human operator is often required to provide on line adjustment, which make the process performance greatly dependent on the experience of the individual operator. It would be extremely useful if some kind of systematic methodology can be developed for the process control model that is suited to kind of industrial process. There are some variables in continuous DCS (distributed control system) suffer from many unexpected disturbance during operation (noise, parameter variation, model uncertainties, etc.) so the human supervision (adjustment) is necessary and frequently. If the operator has a little experience the system may be damage or operated at lower efficiency [1, 4]. One of these systems is the control of turbine speed PI controller is the main controller used to control the process variable. Process is exposed to unexpected conditions and the controller fail to maintain the process variable in satisfied conditions and retune the controller is necessary. Fuzzy controller is one of the succeed controller used in the process control in case of model uncertainties.

But it may be difficult to fuzzy controller to articulate the accumulated knowledge to encompass all circumstance. Hence, it is essential to provide a tuning capability [2, 3]. There are many parameters in fuzzy controller can be adapted. The Speed control of turbine unit construction and operation will be described. Adaptive controller is suggested here to adapt normalized fuzzy controller, mainly output/input scale factor. The algorithm is tested on an experimental model to the Turbine Speed Control System. A comparison between Conventional method and Adaptive Fuzzy Controller are done. The suggested control algorithm consists of two controllers process variable controller and adaptive controller (normalized fuzzy controller).At last, the fuzzy supervisory adaptive implemented and compared with conventional method.

2 BACKGROUND

In refineries, in chemical plants and other industries the gas turbine is a well known tool to drive compressors. These compressors are normally of centrifugal type. They consume much power due to the fact that very large volume flows are handled. The combination gas turbine-compressor is highly reliable. Hence the turbine-compressor play significant role in the operation of the plants. In the above set up, the high pressure steam (HPS) is usually used to drive the turbine. The turbine which is coupled to the compressor will then drive the compressor. The hydraulic governor which, acts as a control valve will be used to throttle the amount of steam that is going to the turbine section. The governor opening is being controlled by a PID which is in the electronic governor control panel. In this paper, it is proposed that the controller be tuned using the Genetic Algorithm technique. Using genetic algorithms to perform the tuning of the controller will result in the optimum controller being evaluated for the system every time. For this study, the model selected is of turbine speed control system.

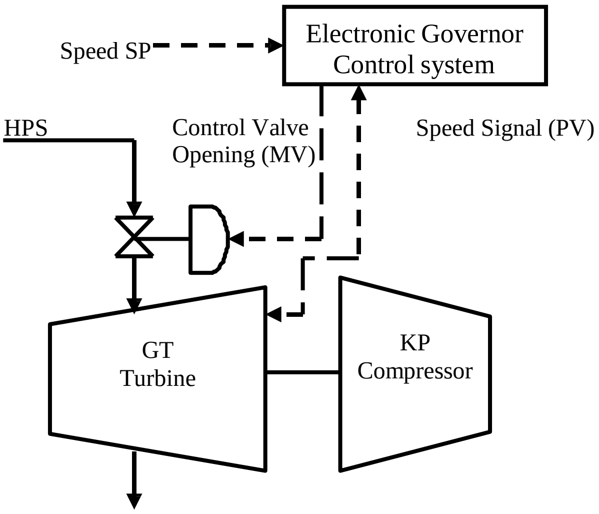


Figure 1: Turbine Speed Control

The reason for this is that this model is often encountered in refineries in a form of steam turbine that uses hydraulic governor to control the speed of the turbine as illustrated above in figure 1. The complexities of the electronic governor controller will not be taken into consideration in this dissertation. The electronic governor controller is a big subject by it and it is beyond the scope of this study. Nevertheless this study will focus on the model that makes up the steam turbine and the hydraulic governor to control the speed of the turbine. In the context of refineries, you can consider the steam turbine as the heart of the plant. This is due to the fact that in the refineries, there are lots of high capacities compressors running on steam turbine. Hence this makes the control and the tuning optimization of the steam turbine significant.

3 EXPERIMENTAL PROCESS IDENTIFICATION

To obtain the mathematical model of the process i.e. to identify the process parameters, the process is looked as a black box; a step input is applied to the process to obtain the open loop time response.

From the time response, the transfer function of the open loop system can be approximated in the form of a third order transfer function:

$$G \ \square \ = \frac{1}{\square \ \square+1} \qquad (1)$$

$$(\square+5)$$

The identified model is approximated as a linear model, but exactly the closed loop is nonlinear due to the limitation in the control signal.

4 PID CONTROLLER

PID controller consists of Proportional Action, Integral Action and Derivative Action. It is commonly refer to Ziegler-Nichols PID tuning parameters. It is by far the most common control algorithm [1]. In this chapter, the basic concept of the PID controls will be explained. PID controller's algorithm is mostly used in feedback loops. PID controllers can be implemented in many forms. It can be implemented as a stand-alone controller or as part of Direct Digital Control (DDC) package or even Distributed Control System (DCS). The latter is a hierarchical distributed process control system which is widely used in process plants such as pharceumatical or oil refining industries. It is interesting to note that more than half of the industrial controllers in use today utilize PID or modified PID control schemes. Below is a simple diagram illustrating the schematic of the PID controller. Such set up is known as non- interacting form or parallel form.

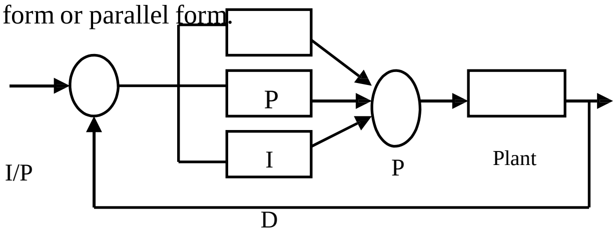


Figure 2: Schematic of the PID Controller – Non-Interacting Form

In proportional control,

$$\text{Pterm} = \text{KP} \times \text{Error} \qquad (2)$$

It uses proportion of the system error to control the system. In this action an offset is introduced in the system.

In Integral control,

$$I_{term} = K1 \times \int \text{Error}\, dt \qquad (3)$$

It is proportional to the amount of error in the system. In this action, the I-action will introduce a lag in the system. This will eliminate the offset that was introduced earlier on by the P-action.

In Derivative control,

$$\square_{\square\square\square m} = K\square x \frac{\square(\square\square\square\square\square)}{\square\square} \tag{4}$$

It is proportional to the rate of change of the error. In this action, the D-action will introduce a lead in the system. This will eliminate the lag in the system that was introduced by the I-action earlier on.

5 OPTIMISING PID CONTROLLER BY CLASSICAL METHOD

For the system under study, Ziegler-Nichols tuning rule based on critical gain K*er* and critical period P*er* will be used. In this method, the integral time Ti will be set to infinity and the derivative time Td to zero. This is used to get the initial PID setting of the system. This PID setting will then be further optimized using the "steepest descent gradient method".

In this method, only the proportional control action will be used. The K*p* will be increase to a critical value K*er* at which the system output will exhibit sustained oscillations. In this method, if the system output does not exhibit the sustained oscillations hence this method does not apply. In this chapter, it will be shown that the inefficiency of designing PID controller using the classical method. This design will be further improved by the optimization method such as "steepest descent gradient method" as mentioned earlier [6].

5.1 Design of PID Parameters

From the response below, the system under study is indeed oscillatory and hence the Z-N tuning rule based on critical gain K*er* and critical period P*er* can be applied. The transfer function of the PID controller is

$$G_c(s) = K_p (1 + T_i (s) + T_d(s)) \tag{5}$$

The objective is to achieve a unit-step response curve of the designed system that exhibits a maximum overshoot of 25 %.

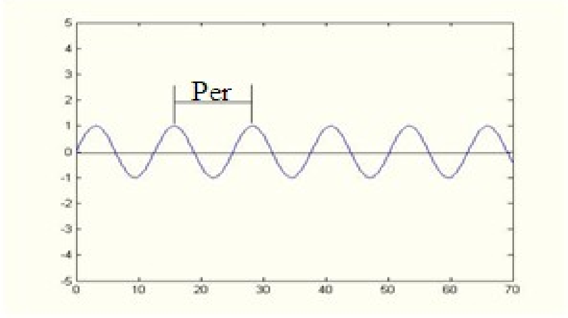


Figure 3: Illustration of Sustained Oscillation

If the maximum overshoot is excessive says about greater than 40%, fine tuning should be done to reduce it to less than 25%.

From Ziegler-Nichols frequency method of the second method [1], the table suggested tuning rule according to the formula shown. From these we are able to estimate the parameters of Kp, Ti and Td.

Controller	Kp	Ti	Td
P	0.5Ker	∞	0
PI	0.45Ker	1 / 1.2 Per	0
PID	0.6 Ker	0.5 Per	0.125 Per

Figure 4: PID Value setting

Consider a characteristic equation of closed loop system

$s^3 + 6s^2 + 5s+ K_p = 0$

From the Routh's Stability Criterion, the value of Kp that makes the system marginally stable can be determined. The table below illustrates the Routh array.

s^3	1	5
s^2	6	Kp
s^1	(30-Kp)/6	0
s^o	Kp	-

With the help of PID parameter settings the obtained closed loop transfer function of the PID controller with all the parameters is given as

$$G_\square(\square) = K_\square(1 + \frac{1}{\square i\square} + \square\square\square)$$

$$= 18 \left(1 + \frac{1}{1.4\square} + 0.3512 \right)$$

$$= \frac{6.3223 \, (\square + 1.4235)^2}{\square} \tag{6}$$

From the above transfer function, we can see that the PID controller has pole at the origin and double zero at s = -1.4235. The block diagram of the control system with PID controller is as follows.

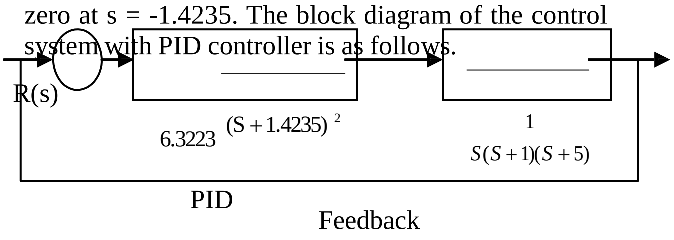


Figure 5: Illustrated Closed Loop Transfer Function

Hence the above block diagram is reduced to

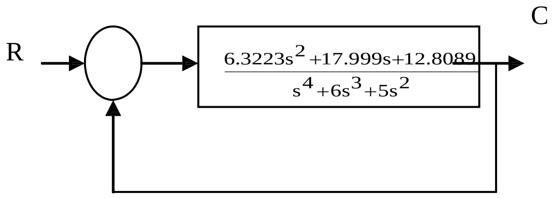


Figure 6: Simplified System

Therefore the overall close loop system response of

$$\frac{C(s)}{R(s)} = \frac{6.3226s^2 + 17.999s + 12.808}{s^4 + 6s^3 + 11.3223s^2 + 18s + 12.8089} \qquad (7)$$

The unit step response of this system can be obtained with MATLAB.

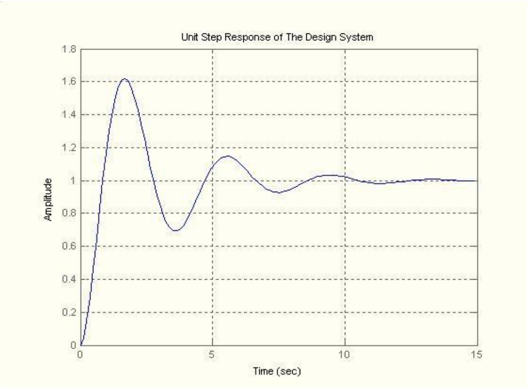


Figure 7: Step Response of Designed System

To optimize the response further, the PID controller transfer function must be revisited. The transfer function of the designed PID controller is

$$G_{\square}(Z) = \frac{\square\square + \square 1Z^{-1} + \square 2Z^{-2}}{1 - Z^{-1}} \qquad (8)$$

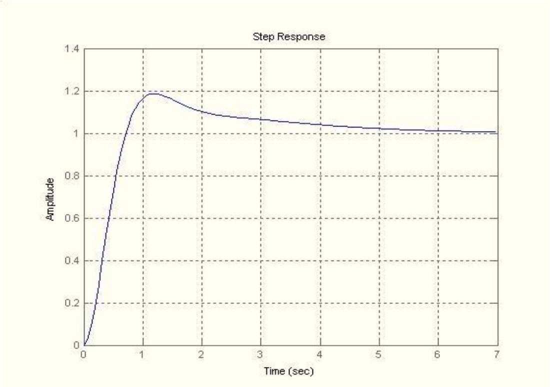


Figure 8: Improved System.

5 OPTIMIZING OF THE DESIGNED PID CONTROLLER

The optimizing method used for the designed PID controller is the “steepest gradient descent method”. In this method, we will derive the transfer function of the controller as the minimizing of the error function of the chosen problem can be achieved if the suitable values of can be determined. These three combinations of potential values form a three dimensional space. The error function will form some contour within the space. This contour has maxima, minima and gradients which result in a continuous surface.

In this method, the system is further optimized using the said method. With the “steepest descent gradient method”, the response has definitely improved as compared to the one in Fig. 9 (a). The settling time has improved to 2.5 second as compared to 6.0 seconds previously. The setback is that the rise time and the maximum overshoot cannot be calculated. This is due to the “hill climbing” action of the steepest descent gradient method. However this setback was replaced with the quick settling time achieved. Below is the plot of the error signal of the optimized controller. In the figure below it is shown that the error was minimized and this correlate with the response shown in Figure 9(b).

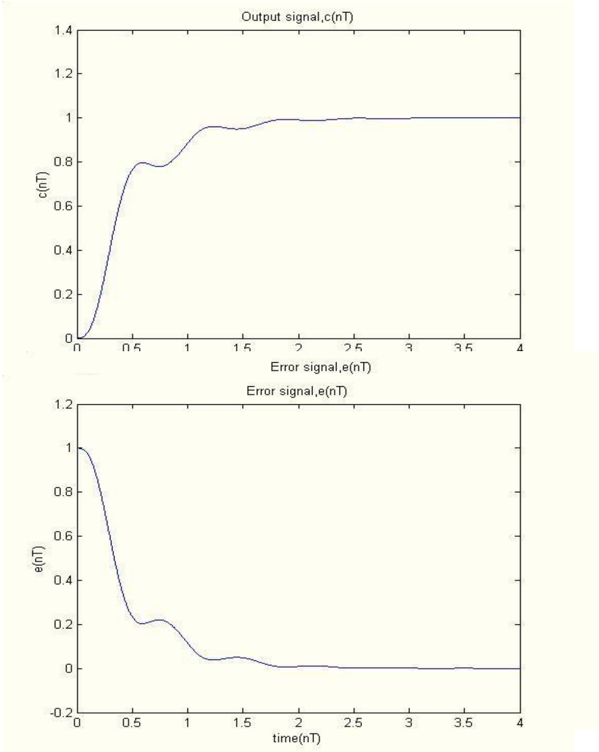


Figure 9 (a) & (b): Optimization of Steepest Descent Gradient Method & Error Signal

From the above figure, the initial error of 1 is finally reduced to zero. It took about 2.5 to 3 seconds for the error to be minimized.

6 IMPLEMENTATION OF ADAPTIVE FUZZY CONTROLLER ON EXPERIMENT CASE STUDY

6.1 Normalized Fuzzy Controller

To overcome the problem of PID parameter variation, a normalized Fuzzy controller with adjustable scale factors is suggested. In our experimental case study, the fuzzy controller designed has the following parameters:

- Membership functions of the input/output signals have the same universe of discourse equal to 1
- The number of membership functions for each variable is 5 triangle membership functions denoted as NB (negative big), NS (negative small), Z (zero), PS (positive small) and PB (positive big) as shown in Fig. 10.

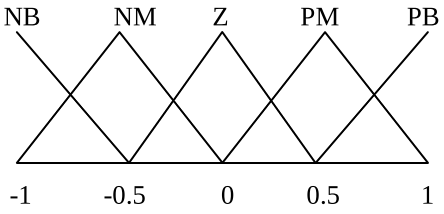


Figure 10: Normalized membership function of inputs and output variables

- Fuzzy allocation matrix (FAM) or Rule base as in Table1.

Table 1: FAM Normalized Fuzzy Controller

Δe e	NB	NM	Z	PM	PB
NB	PB	PB	PM	Z	Z
NM	PM	PB	PM	Z	Z
Z	PM	PM	Z	NM	NM
PM	Z	Z	NM	NB	NB
PB	Z	NM	NB	NB	NB

- Fuzzy inference system is mundani.
- Fuzzy inference methods are “min” for AND, “max”for OR, “min” for fuzzy implication, “max” for fuzzy aggregation (composition), and “centroid” for Defuzzification.

Adjusting the gains according to the simulation results, the system responses for different input/output gains are shown in Fig. 11.

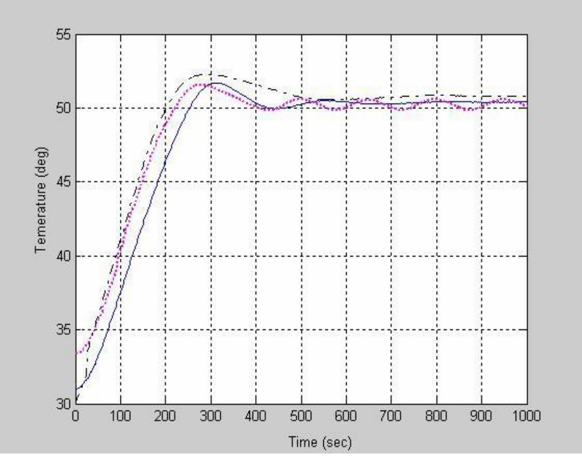


Figure 11: Actual responses for different input output gains

From the analysis of the above responses, we can conclude that:

- Decreasing input scale factors increase the response offset.
- Increasing output scale factor fasting the response of the system but may cause some oscillation.

So the selection must compromise between input and output scale factors.

In the following section we try to adapt the output scale factor with constant input scale factor at 10 error scale, and 15 rate of error scale based on manual tuning result. There are two method tested to adapt the output scale factors, GD (Gradient Decent) adaptation method and supervisor fuzzy.

6.2 Fuzzy Supervisory Controller

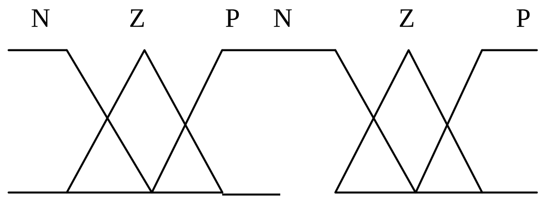
In this method I try to design a supervisor fuzzy controller to change the scale factors online design of the supervisor can be constructed by two methods:

a) Learning method
b) Experience of the system and main requirements must be achieved.

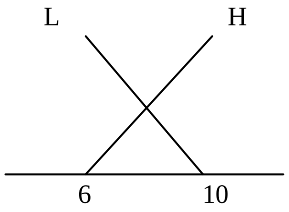
In this paper, the supervisor controller is built according to the accumulative knowledge of the previous tuning methods.

The supervisor fuzzy controller has the following parameters:

- The universe of discourse of input and output is selected according to the maximum allowable range and that is depend on process requirements
- The number of membership functions for input variables is 3 triangle membership functions denoted as N (negative), Z (zero) and P (positive). For output variable is 2 membership functions denoted as L (low) and H (High) as shown in Fig, 12.



a) Error b) rate of error



c) Output Scale Factor

Figure 12: Membership Function of Inputs and Output of supervisory fuzzy control

- Fuzzy allocation matrix (FAM) or rule base as in Table 2.

Table 2: FAM of Supervisory Fuzzy Controller

Δe e	N	Z	P
N	H	H	L
Z	L	L	H
P	L	H	H

- Fuzzy Interference system is mundani.
- Fuzzy Inference methods are “min” for AND, “max” for OR, “min” for fuzzy implication, “max” for fuzzy aggregation (composition), and “centroid” for Defuzzification.

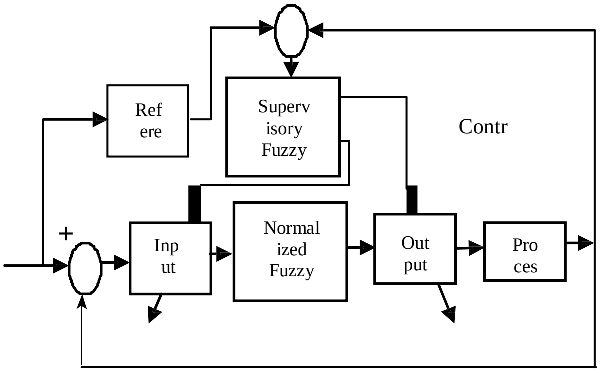


Figure 13: Supervisory Fuzzy Controller

Firstly, we supervise the output gain only as in GD method to compare between them. Reference model is a unity gain. Fig. 14 shows the system response using supervisory fuzzy controller. The two responses are almost similar. The response of supervisor fuzzy is relatively faster. Tuning both input and output scale factors using supervisor controller, the supervisor fuzzy will be multi-input multi-output fuzzy controller without coupling between the variables, i.e. the same supervisor algorithm is applied to each output individually with different universe of discourses.

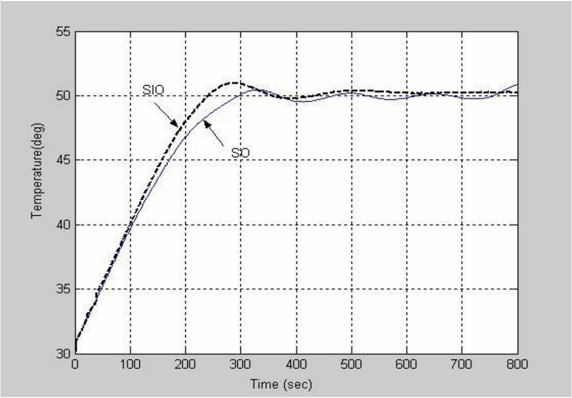


Figure 14: System responses for single and multi-output supervisor

All the previous results are taken with considering that the reference response is step. In practice, there is no physical system can be changed from initial value to final value in now time. So, the required performance is transferred to a reference model and the system should be forced to follow the required response (overshoot, rise time, etc.). The desired specification of the system should to be: overshoot≤ 20%; rise time ≤ 150sec; based on the experience of the process. The desired response which achieves the desired specification is described by equation.

$$y_d(t)=A*[1-1.59e^{-0.488t}\sin 0.3929t+38.83*\pi/180)] \quad (9)$$

Where A: step required. Fig. 15 compares between the two responses at different values and reference model response. This indicates a good responses and robustness controller.

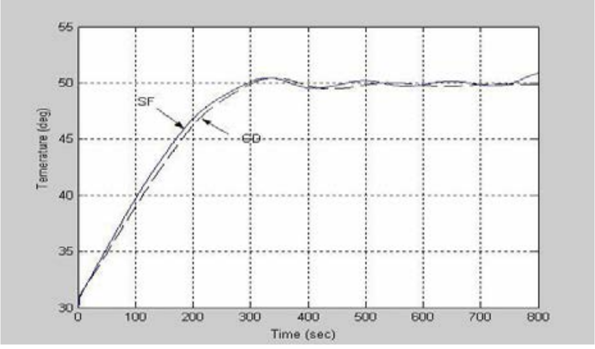


Figure 15: Analysis of Steepest gradient & Adaptive Fuzzy Method

8 RESULTS OF IMPLEMENTED ADAPTIVE FUZZY CONTROLLER

In the following section, the results of the implemented Adaptive Fuzzy Controller will be analyzed [4]. The Adaptive Fuzzy designed PID controller is initially initialized and the response analyzed. The response of the

Adaptive Fuzzy designed PID will then be analyzed for the smallest overshoot, fastest rise time and the fastest settling time. The best response will then be selected.

From the above responses fig 15, the Adaptive Fuzzy designed PID will be compared to the Steepest Descent Gradient Method. The superiority of Adaptive Fuzzy Controller against the SDG method will be shown. The above analysis is summarized in the following table.

Table 3: Results of SDGM Designed Controller and Adaptive Fuzzy Designed Controller.

Measuring Factor	SDGM Controller	AF Controller	% Improvement
Rise Time	10	0.592	40.8
Max. Overshoot	NA	4.8	NA
Settling Time	2.5	1.66	33.6

From Table 3, we can see that the Adaptive Fuzzy designed controller has a significant improvement over the SDGM designed controller. However the setback is that it is inferior when it is compared to the rise time and the settling time. Finally the improvement has implication on the efficiency of the system under study. In the area of turbine speed control the faster response to research stability, the better is the result for the plant.

9 CONCLUSION

In conclusion the responses had showed to us that the designed PID with Adaptive Fuzzy Controller has much faster response than using the classical method. The classical method is good for giving us as the starting point of what are the PID values. However the approached in deriving the initial PID values using classical method is rather troublesome. There are many steps and also by trial and error in getting the PID values before you can narrow down in getting close to the “optimized” values. An optimized algorithm was implemented in the system to see and study how the system response is. This was achieved through implementing the steepest descent gradient method. The results were good but as was shown in Table 3 and Figure 15. However the Adaptive Fuzzy designed PID is much better in terms of the rise time and the settling time. The steepest descent gradient method has no overshoot but due to its nature of “hill climbing”, it suffers in terms of rise time and settling time. With respect to the computational time, it is noticed that the SDGM optimization takes a longer time to reach it peak as compare to the one designed with GD. This is not a positive point if you are to implement this method in an online environment. It only means that the SDGM uses more memory spaces and hence take up more time to reach the peak. This paper has exposed me to various PID control strategies. It has increased my knowledge in Control Engineering and Adaptive Fuzzy Controller in specific. It has also shown me that there are numerous methods of PID tunings available in the academics and industrial fields.

10 REFERENCES

[1] Astrom, K., T. Hagglund: PID Controllers; Theory, Design and Tuning, Instrument Society of America, Research Triangle Park, 1995.

[2] M. A. El-Geliel: Supervisory Fuzzy Logic Controller used for Process Loop Control in DCS System, CCA03 Conference, Istanbul, Turkey, June 23/25, 2003.

[3] Kal Johan Astroum and Bjorn Wittenmark: Adaptive control, Addison-Wesley, 1995

[4] Yager R. R. and Filer D. P.: Essentials of Fuzzy Modeling and Control, John Wiley, 1994.

[5] J. M. Mendel: Fuzzy Logic Systems for Engineering: A tutorial, Proc. IEEE, vol. 83, pp. 345-377, 1995.

[6] L. X. Wang: Adaptive Fuzzy System & Control design & Stability Analysis, Prentice-Hall, 1994.