DESIGN OF AN ADAPTIVE CONSTRAINED BASED NEURO-FUZZY CONTROLLER FOR FAULT DETECTION OF A POWER PLANT SYSTEM

Priya R
Research Scholar, University of Kerala,
Trivandrum, Kerala 695581, India
priyanil2007@gmail.com

Elizabeth Sherly
Professor, Indian Institute of Information Technology and Management-Kerala,
Trivandrum, Kerala 695581, India
sherly@iiitmk.ac.in

Abstract: This paper proposes an adaptive constraint based framework for fault detection of a complex thermal power plant system. In many complex systems, representation of precise and crisp constraints uses formal specification languages such as Object Constraint Language (OCL). Here, a constraint based neuro-fuzzy controller to tackle imprecise constrained objects is proposed. The proposed inference system is used to identify the intensity of faults by mapping instance values against the constraints. The imprecise constraints defined as fuzzy constraints are prioritized using fuzzy weights assigned to rules. Back-propagation algorithm is used to train and calibrate the controller to capture the dynamic behavior of the system. The system is adopted into a coal-fired Thermal power plant system, which is controlled by different parameters in a constrained environment. The output of the neuro-fuzzy inference system is compared against actual plant site alarm that functions based on rule based system and has been found that the intensity of faults can be accurately determined.

Keywords: Fuzzy inference systems; fuzzy control; neuro-fuzzy systems; process control; fault detection and diagnosis.

I. Introduction

Unified Modeling Language (UML) is widely used for structural modeling of any complex object oriented system, with the help of Object Constraint Language(OCL). OCL is a powerful formal specification language that represents constraints like invariants on classes and types, preconditions and postconditions on operations. However, representing all the constraints of a complex system in OCL requires a rule base system that is tedious and exhaustive. Moreover, the constraints need to be very precise to be represented using OCL. For incorporating imprecise constraints in an intelligent manner, fuzzy control technique can be used. If the constraints of the complex system are of different priority levels, they are mapped to weighted fuzzy rules of the fuzzy control system.

Recently, the integration between fuzzy logic and neural network known as neuro-fuzzy or the fuzzy neural network (FNN) has drawn attention of many researchers in the field of control engineering [1]. Combining the learning ability of neural network and human-like reasoning of fuzzy logic makes FNN a very flexible intelligent control technique for many control applications. In this paper, a model is proposed for detection of faults of a constrained complex system, whose constraints are represented as fuzzy rules. The scopes for adopting artificial intelligent based modeling for fault detection in control systems are to increased accuracy and reliability. The proposed neuro-fuzzy model is adapted to the Boiler subsystem of a Coal-fired Thermal Power Plant that helps to achieve generating the rule parameters automatically using neural network. Hence, constraint representation of complex systems using a neuro-fuzzy controller doubles the advantages of the cognitive thinking capability of fuzzy logic and learning capability of neural networks. The rest of the paper is organized as follows. Section II gives a brief summary on the various works carried out using neuro-fuzzy soft computing technique. Section III explains the methodology of the proposed work. The methodology is tested on a complex system of Thermal power plant and it is discussed in Section IV. The performance of the model is evaluated and the work is concluded with scope for future improvements in Section V.

II. Literature Survey

Gurpreet S. Sandhu and Kuldip S. Rattan explained how to design and train a proportional neuro-fuzzy controller. The architecture and the learning algorithm of a proportional fuzzy controller (PFLC) is presented. A step by step algorithm for the off-line training of a PFLC is demonstrated [2]. A survey on Neuro-fuzzy systems was done by José Vieira, Fernando Morgado Dias and Alexandre Mota. The article summarizes the architecture of popular neuro-fuzzy systems.[3]. Disha Sharma has explained the fuzzy modeling problem and the steps involved in it. The paper provides an introduction to the use of fuzzy sets and fuzzy logic for the approximation of functions and modeling of static and dynamic systems [4]. A neuro-fuzzy designer control algorithm using neuro-fuzzy identifier is proposed by Minho Lee *et al.* which could be used for modelling fuzzy system whose rules are uncertain.[5]. The authors in [6] develop a fuzzy control algorithm for a warm water plant. There are many other applications[7] of conventional fuzzy control, including robot [8], steel furnace [9], waste water treatment [10] and so on.[11],[12],[13],[14],[20]. In addition, fuzzy control has been widely used in various consumer electronic devices such as video cameras, washing machines, TV, and sound systems in the late 1980sand early 1990s [15], [16]. The authors in [17] have proved that input parameter selection is very critical in ANFIS learning.

The above literature enforces the fact that neuro-fuzzy control systems have been widely used in a variety of control applications for improved performance. In this work, fault is detected from the model of a constrained complex system with imprecise constraints, by modeling constraints as fuzzy rules. The model is trained for accuracy by incorporating the learning capability of neural network.

III. An adaptive Fuzzy logic controller for fault detection of a constrained complex system

Fuzzy modeling of complex control systems without prior knowledge of fuzzy rules often requires fine tuning of fuzzy controller. The learning capability of neural network facilitates fine tuning of design parameter of fuzzy logic. Weights are assigned to the fuzzy rules based on the priority of constraints of the system. The proposed fuzzy model helps to detect fault intensity when it is validated against input parameters. The architecture of the model for fault intensity detection is shown in Figure (1).

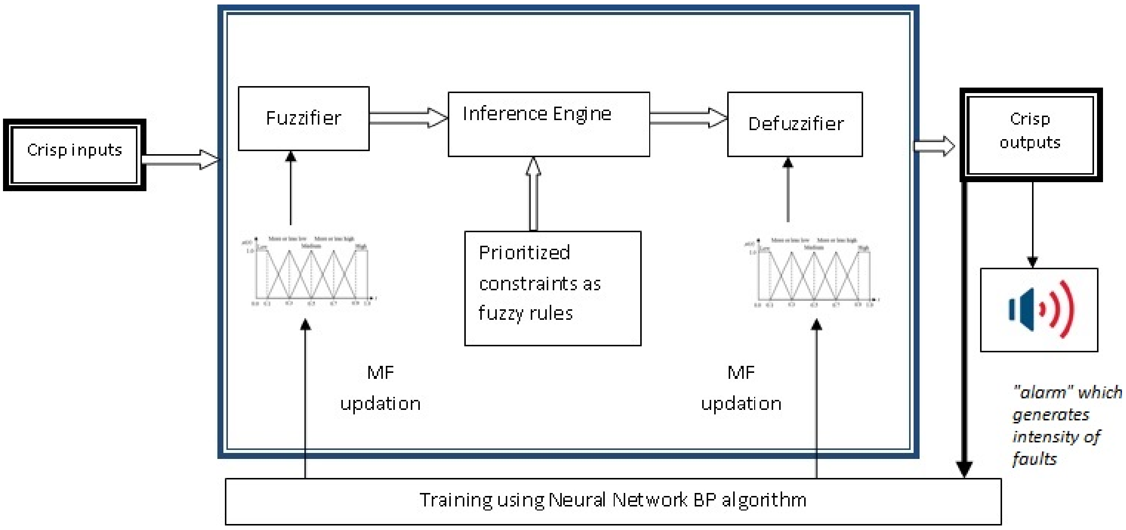

Figure 1. Architecture of the proposed model for fault detection of complex systems using neurofuzzy controller

The transformation of raw data to fuzzy value using membership functions marked the beginning of fuzzification. The main issue is to generalize the parameters of membership functions for fuzzification from the training examples to the entire domain by training the neural network.

Let F_{NN} denote the function directly encoding the network operation. The network is trained so that F_{NN} approximates f,

$$F_{NN} \approx f \qquad (1)$$

where f is the function that gives the most optimal alarm values. For the purpose of training, a set of training examples are selected from the domain. Each example is represented by (x,y) where y=f(x). The ideal goal is that after training,

$$y = f(x) = F_{NN}(x) \qquad (2)$$

for every x in the domain, not limited to the training set. Sum of squared error is most commonly used to calculate accuracy of F_{NN} .

$$SSE = \sum_{i=1}^{n} (x_i - y_i)^2 \quad (3)$$

SSE can be a high value because of mainly the following four cases of errors in parameters of membership functions (µ1 and µ2) of fuzzification process.

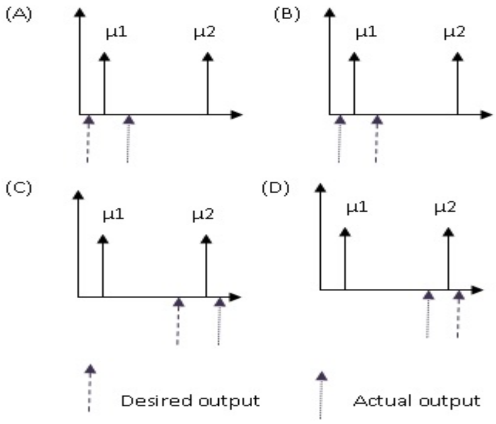

Figure 2. Four possibilities of errors

(1) Case A- The desired value is located to the left of µ1, but the actual output lies between µ1 and µ2.

(2) Case B - The desired value is located between µ1 and µ2, but the actual output lies to the left of µ1.

(3) Case C - The desired value is located between µ1 and µ2, but the actual output lies to the right of µ2.

(4) Case D - The desired value is located to the right of µ2, but the actual output lies between µ1 and µ2.

In all the cases the output value is being produced by incorrect fuzzy sets, and hence the shape of input membership functions needs to be adjusted. The design of the network architecture and the training algorithm is aimed at improving this capability of reducing the squared error.

The architecture of the proposed model for training can be compared with the architecture of a neural network where each layer of the neural network corresponds to each step of designing a fuzzy inference system. The architecture consists of input layer, which corresponds to receiving the actual crisp value input. The next hidden layer is the fuzzification layer that uses the input membership functions to convert the crisp inputs to fuzzy inputs. This layer is followed by the layer with neurons R_i, which are the rules or the constraints of the complex system. The next layer is the one with the function of defining the output membership function. The output layer uses defuzzification method to convert all the fuzzy outputs obtained to one crisp value.

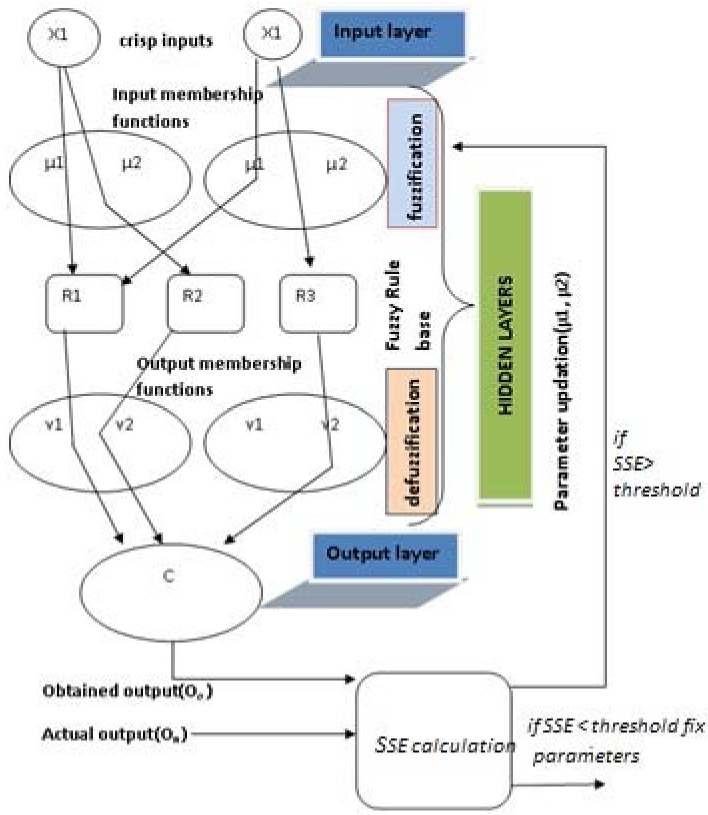

Figure 3. Training the fuzzy controller by back-propagating the error to the neural-like architecture.

The X-, R-, and C-modules are the neurons in a layered neural network and the μ - and v-values as the adaptable weights of the network. This corresponds to the parameters passed to the membership functions during the fuzzification stage and the defuzzification stage of designing a fuzzy inference system. The X-module layer is the input layer of a multi-input neural network whereas the C-module layer is the output layer. The R-module layer serves as the hidden or intermediate layer that defines the fuzzy rule base. The fact that one μ -module can be connected to more than one R-module is equivalent to the connections in a neural network. This is of key importance for keeping the structural integrity of the fuzzy controller intact.

For training a neural network, back-propagation algorithm is used. For each epoch, error value is calculated as per Equation (3). If the error value is less than threshold, the parameters of membership functions are fixed. If the error value is greater than threshold, then parameters are changed and the training continues, by distributing back the control value among the control rules. The training continues until the error becomes less than threshold or until the desired number of epochs are reached. The main application of neural network is for the fine tuning of the parameters of the membership functions as a reverse mechanism deduced from the forwarding inference machinery of the fuzzy logic controller.

The selection of parameters of the membership functions of fuzzification governs the rules to be triggered. An inappropriate parameters fixing of membership functions result in wrong selection of the rules to be triggered which in turn results in the generation of a wrong crisp output. This deteriorates the performance of the model when the model finds applications in areas like fault tolerance. Hence in this work, the accuracy of the parameters are judged and fixed based on the calculated SSE.

Algorithm 1: for training the fuzzy control system using neural network.

Input : .DS_T, MF_N ,MF_T,μ1, μ2, E_T, N_E
Output : μ1', μ2'

Step 1 : Define the number of membership functions(MF_N) and type of membership functions (MF_T) for fuzzification of input variables.

Step 2 : Define the initial parameters for membership functions. (μ1 and μ2)

Step 3 : Define the training data set.(DS_T)

Step 4 : Set the error tolerance value or the threshold value.(E_T)

Step 6 : Define the number of epochs(N_E).

Step 7 : # Epochs =1.

Step 8 : Compute E_A = SSE based on Equation (3).

Step 9 : If the error $E_A > E_T$ go to step 10, otherwise go to step 13.

Step 10 : #Epochs = # Epochs + 1.

Step 11 : Change the parameters of the membership functions by training the system using back-propagation algorithm .

Step 12 : Repeat Step 8 – Step 11 until

($E_A < E_T$ or # Epochs > N_E or E_A becomes a constant)

Step 13 : Take the paramaters of μ1 and μ2 as the new parameters μ1' and μ2' for defining the fuzzy membership function.

A fuzzy inference system can be designed with the new parameters μ1' and μ2' for the membership function of the fuzzification process of the input variables. The fuzzified values of the crisp inputs trigger one or more rules from the fuzzy rule base of the inference system. All the rules of the fuzzy rule base are assigned weights based on the priority of corresponding constraints. This model with fuzzy rules and membership functions identified using neural network can be used for fault detection when the model is validated against instance values, with great accuracy.

IV. Simulation results for fault detection in a thermal power plant system

The system taken for study is a coal-fired thermal power plant. The primary fuel fed to the power plant is coal, which is used to generate electricity. The thermal power plant is a complex system and it is constrained with a set of parameters. Coal is fed to the pulverizer component and is pulverized to fine granules. This is given to the boiler component. The steam generated from the boiler rotates a turbine to which a generator is attached. This generates electricity.

Literature reveals that many mathematical coal mill modeling techniques have been proposed and implemented which resulted in optimized milling process[18]. But in this work, an attempt is made to optimise an object oriented model of a coal mill model by using an optimal fuzzy logic controller to represent the fuzzy constraints of the system. The proposed model has the advantage of simulation and testing of the model before actually building the system.

In this work, the subsystem taken for study is "Boiler", and the three dependent parameters of Boiler are "Steam_Inlet_temperature", "Steam_Inlet_Pressure" and "Feed_Water_Temperature". These parameters are constrained to a set of domain values. In a fuzzy controlled system, faults of different intensities can be identified based on the values of these parameters. Table (1) shows the permissible range for these parameters.

Table 1 . Domain range of three input parameters for Boiler

Input parameters	Minimum	Maximum
Steam_Inlet_temperature	320	340
Steam_Inlet_Pressure	20	30
Feed_Water_Temperature	235	255

The crisp value of these parameters fed to the model needs to be validated against the constraints defined as fuzzy rules. The rule selection is on the basis of the parameters of membership functions. This ultimately governs the intensity of faults detected. Hence choice of parameters for membership function is very crucial.

In this paper, neural network training algorithm of back-propagation is used to identify the most optimal parameters for input and output membership functions. For accurate parameter identification, the control system is trained with training inputs as explained in the Algorithm1.

Triangular membership functions shown in Equation (4) gives the best drive performance while implementing fuzzy logic in control systems.[19].Hence, the input variable of Steam_Inlet_temperature, Steam_Inlet_Pressure and Feed_Water_Temperature are defined with three triangular membership functions.

$$\text{triangle}(x; a, b, c) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \tag{4}$$

The linguistic variables associated with each input parameter are LO(Low), RT(Right) and HI(High). Table (2) shows the parameter range for the linguistic variables associated with each input.

Table 2 . Parameter range of input linguistic variables

Ling.Var	**Parameters**		
	St_Inlet_Temp	St_Inlet_Pr	Feed_Water_Temp
LO	[320 330]	[20 24]	[235 245]
RT	[325 335]	[23 27]	[240 250]
HI	[330 340]	[26 30]	[245 255]

Training data set with expected alarm values obtained from thermal power plant site is used to train the model for many epochs. Increase in the number of epochs increases the accuracy of the model as the average training error decreases. It is seen that as the number of epochs increases, the training error decreases and remains unchanged after a specific epoch.

Table 3. Sample training dataset

No.	St_Inlet_Temp	St_Inlet_Pr	Feed_Water_Temp	Alarm
1	320	20	235	-1.5
2	330	26	246	0
3	340	30	255	1.5
4	336	28	252	1.3
5	334	26.5	250	1
6	332	27	248	0
7	325	22	240	-1.3
8	338	28	250	1.2
9	335	27	248	0.98
10	329	25	244	-0.98

The output linguistic variable “alarm” and membership function are defined as shown in Table(4) and Figure(4) respectively.

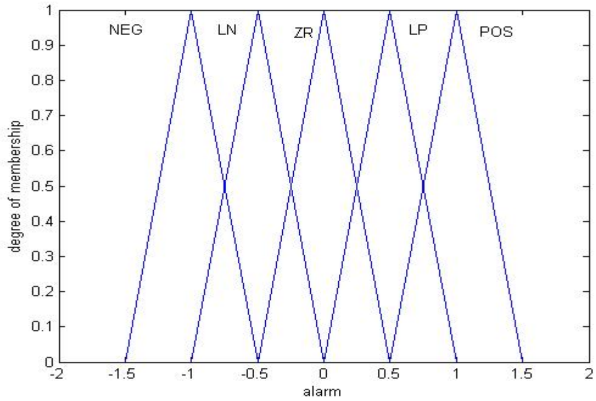

Figure 4. Output membership function graph

Table 4. Range of output linguistic variable “alarm”

Alarm(output variable)		
NEG	-1.5	-0.5
LN	-1	0
ZR	-0.5	0.5
LP	0	1
POS	0.5	1.5

Fuzzy rules are defined with the three parameters as antecedents and alarm as the consequent, shown in Tables 5a, 5b and 5c.

Table 5a. Fuzzy rules when St_Inlet_Temp=LO

St_Inlet_Temp=LO			
St_Inlet_Pr	F-W-temp		
	LO	RT	HI
LO	NEG	LN	LN
RT	LN	LN	LN
HI	LN	LP	LP

Table 5b. Fuzzy rules when St_Inlet_Temp=HI

St_Inlet_Temp=HI			
St_Inlet_Pr	F-W-temp		
	LO	RT	HI
LO	LN	LP	LP
RT	LP	LP	LP
HI	LP	LP	POS

Table 5c.Fuzzy rules when St_Inlet_Temp=RT

St_Inlet_Temp=RT			
St_Inlet_Pr	F-W-temp		
	LO	RT	HI
LO	LN	LN	LP
RT	LN	ZR	LP
HI	LP	LP	LP

The defuzzification uses the weighted average method given in Equation(5) and the results are shown in Table (6).

$$x_{final} = \frac{x_1\mu_1 + x_2\mu_2 + \ldots + x_n\mu_n}{\mu_1 + \mu_2 + \ldots + \mu_n} \tag{5}$$

Table 6.Comparison of expected and obtained alarm values

St_Inlet_Temp	St_Inlet_Pr	F-W-Temp	Alarm (Fuzzy controller) x	Alarm (Expected Result) y
326	23	242	-1	-1.3
330	25	245	0	0
336	29	252	1	1.3
330	26	246	0.146	0
336	28	252	1	1.3
334	26.5	250	0.751	1

The sum of squared obtained is 0.3533, which is not a negligible value.

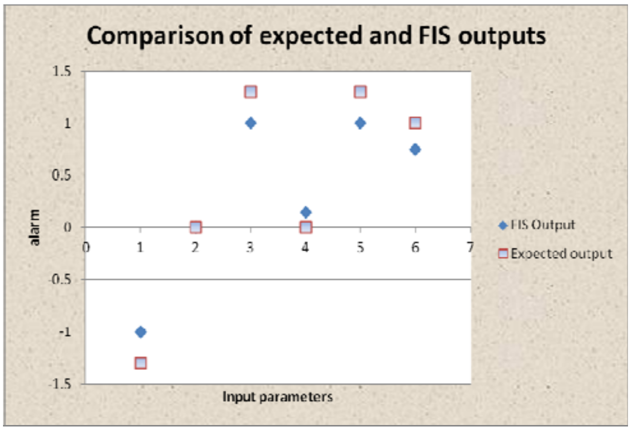

Figure 5. Graph showing comparison of expected and obtained outputs.

The high value of SSE reveals that the membership parameters are not optimal. Hence the FIS is trained using the back-propagation algorithm of neural network. Figure(6) shows the architecture of training FIS of Boiler with three parameters and twenty seven rules.

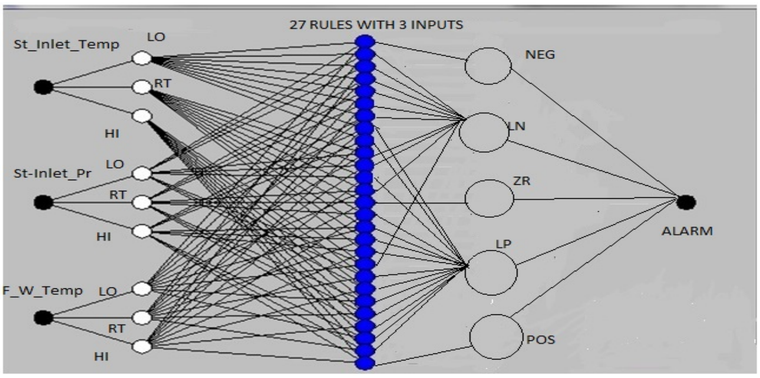

Figure 6.Architecture for training the FIS of Boiler with three parameters

Table 7 . Training error through epochs

Epochs	MSE
10	1.055
20	0.999
50	0.837
100	0.599
200	0.298
400	0.142
500	0.107

MSE decreases with increase in epchs and the corresnding values is given in Table 7. A graphical reresentation is shown in Figure(7).

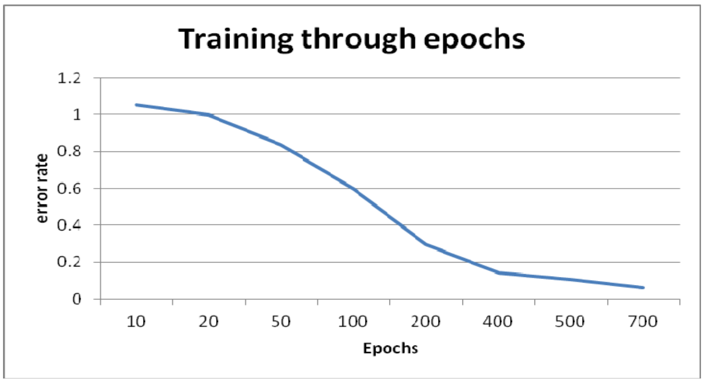

Figure 7. A graphical representation of decrease in error rate with epochs

Table 8. Comparison of expected and obtained alarm values

St_Inlet_Temp	St_Inlet_Pr	F-W-Temp	Alarm (Trained Fuzzy controller) x	Alarm (Expected Result) y
320	20	235	-1.5	-1.5
330	26	246	-0.037	0
340	30	255	1.5	1.5
336	28	252	1.22	1.3
334	26.5	250	1.06	1
332	27	248	0.078	0
325	22	240	-1.31	-1.3
338	28	250	1.27	1.2
335	27	248	0.857	0.98
329	25	244	-0.958	-0.98

The SSE value now reduced to 0.038, which is a negligible value.

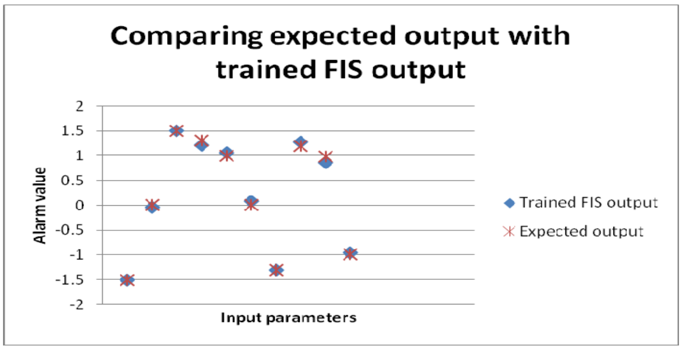

Figure 8: Graph showing comparison of expected and obtained outputs.

For a constrained complex system, the constraints or the rules may be of different priority levels. Here, the priority of constraints is mapped to the weights of fuzzy rules in the fuzzy rule base. The priority *p* of rule *i*, *p(i)* is defined based on Equation(6).

$$p\ (i) = \begin{cases} 1 & iff\ consq(rule) = \{NEG, POS\} \\ 0 < x < 1 & iff\ consq(rule) = \{POS, LP, LN\} \\ 0 & iff\ consq(rule) = \{ZR\} \end{cases} \quad \textbf{(6)}$$

The defuzzification is based on equation(7), which is a modified weighted average method with priority *p(i)*.

$$x_{final} = \frac{\sum_{i=1}^{n} x_i \mu_i pr_i}{\sum_{i=1}^{n} x_i \mu_i} \quad \textbf{(7)}$$

Table 9. Comparison of expected, Trained FIS and prioritised trained FIS output values

St_Inlet_ Temp	St_Inlet_Pr	F-W-Temp	Alarm(Trained Fuzzy controller) x	Alarm (Expected Result) y	Alarm (Trained Prioritised fuzzy controller)
320	20	235	-1.5	-1.5	-1.5
330	26	246	-0.037	0	-0.004
340	30	255	1.5	1.5	1.5
336	28	252	1.22	1.3	1.3
334	26.5	250	1.06	1	1.01
332	27	248	0.078	0	0.04
325	22	240	-1.31	-1.3	-1.33
338	28	250	1.27	1.2	1.21
335	27	248	0.857	0.98	1.01
329	25	244	-0.958	-0.98	-0.98

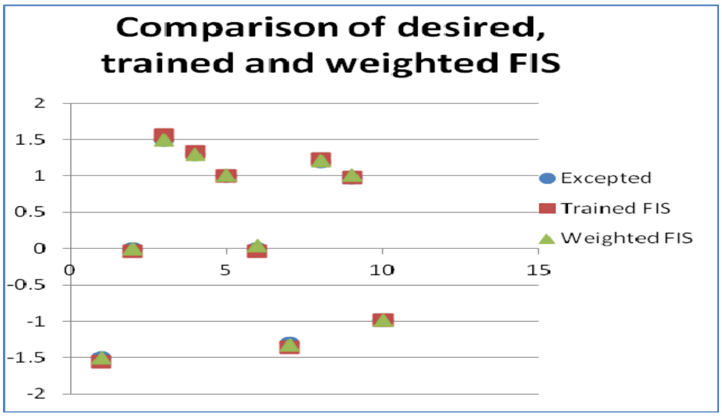

Figure 9: Graph showing comparison of expected, trained and weighted(prioritised) outputs.

The SSE value on trained prioritized fuzzy controller, gave 0.0037, which is a negligible value. Hence, the parameters of membership functions for each of the input linguistic variable are considered the optimal. The above results reveal that a more accurate alarm value is obtained from the fuzzy control system with weighted fuzzy rules.

V. Conclusion

A fuzzy controller is designed and implemented for fault detection of Boiler subsystem of a coal-fired thermal power plant system. A rule based system can be used for fault detection if the rules are crisp and precise in nature. We have proposed and implemented an intelligent fuzzy controller for handling imprecise and uncertain situations and proved its suitability for fault detection of a constrained complex system. The unknown dynamics of fuzzy rules are identified using the soft computing technique of neural networks. The priority of constraints of the complex systems defines the weights of the fuzzy rules in the rule base. The results reveal that the prioritised and trained fuzzy control system can be used for generating alarms of different intensities from -1.5 to 1.5, based on which the intensity of the fault can be identified. This control system can replace the existing rule based system which generates only one specific alarm for any constraint violated. The control result can be improved by resizing the fuzzy sets and finer tuning for the membership functions.

References

[1] Lee, Ching-Hung, and Ching-Cheng Teng. "Identification and control of dynamic systems using recurrent fuzzy neural networks." IEEE Transactions on fuzzy systems 8.4 (2000): 349-366.

[2] Rattan, Kuldip S., T. Brehm, and Gurpreet S. Sandhu. "Analysis and Design of a Proportional Fuzzy Logic Controller." Proceedings of the International Fuzzy Systems and Intelligent Control Conference. 1994.

[3] Vieira, Jose, F. Morgado Dias, and Alexandre Mota. "Neuro-fuzzy systems: a survey." 5th WSEAS NNA International Conference. 2004.

[4] Sharma, Disha. "Designing and modeling fuzzy control Systems." International Journal of Computer Applications 16.1 (2011): 46-53.

[5] Lee, Minho, Soo-Young Lee, and Cheol Hoom Park. "Neuro-fuzzy identifiers and controllers." Journal of Intelligent & Fuzzy Systems 2.1 (1994): 1-14.

[6] W. J. M. Kickert and H. R. Van Nauta Lemke, "Application of a fuzzy logic controller in a warm water plant," Automatica, vol. 12, pp.301–308, (1976).

[7] Snehal N. Rajguru, Pritesh Shah, NeelaRayavarapu and PriyankaTupe, "Application of fuzzy controllers for flow control processes in chemical industries" Journal of Chemical and Pharmaceutical Research, 6(12):pp.403-10 Research Article ISSN : 0975-7384 CODEN(USA) : JCPRC5, (2014).

[8] I. Baturone, F. J. Moreno-Velo, S. Sanchez-Solano, and A. Ollero, "Automatic design of fuzzy controllers for car-like autonomous robots," IEEE Trans. Fuzzy Syst., vol. 12, no. 4, pp. 447–465, (Aug. 2004).

[9] R. J. Kornblum and M. Tribus, "The use of Bayesian inference in the design of an endpoint control system for the basic oxygen steelfurnace," IEEE Trans. Syst., Man, Cybern., vol. SMC-6, no. 2, pp. 339–348, (Mar./Apr. 1970).

[10] R. M. Tong, M. B. Beck, and A. Latten, "Fuzzy control of the activated sludge wastewater treatment process," Automatica, vol. 6, pp. 695–701,(1980).

[11] Om Prakash Verma and Himanshu Gupta, "Fuzzy Logic Based Water Bath Temperature Control System" Volume 2, Issue 4, International Journal of Advanced Research in Computer Science and Software Engineering Research Paper, (April 2012).

[12] Isizoh A. N., Okide S. O, Anazia A.E.. Ogu C.D. "Temperature Control System Using Fuzzy Logic Technique".(IJARAI) International Journal of Advanced Research in Artificial Intelligence, Vol. 1, no. 3, (2012).

[13] Snejana Yordanova, Daniel Merazchiev, and Lakhmi Jain "A Two-Variable Fuzzy Control Design With Application to an Air-Conditioning System" IEEE transactions on fuzzy systems, Vol. 23, no. 2, April 2015.

[14] Kapil Dev Sharma, M. Ayyub, Sumit Saroha, Ahmad Faras "Advanced Controllers Using Fuzzy Logic Controller (FLC) for Performance Improvement" International Electrical Engineering Journal (IEEJ) Vol. 5 no.6, pp. 1452-58,(2014).

[15] T. Haruki and K. Kikuchi, "Video camera system using fuzzy logic," IEEE Trans. Consumer Electron., vol. 38, no. 3, pp. 624–634, (Aug.1992).

[16] Y. M. Lee, S. I. Jang, K. W. Chung, D. Y. Lee, W. C. Kim, and C. W.Lee, "A fuzzy-control processor for automatic focusing," IEEE Trans.Consumer Electron., vol. 40, no. 2, pp. 138–144,(May 1994).

[17] Jyh-Shing Roger Jang. Input Selection for ANFIS Learning, Proc.5th IEEE International Conference on Fuzzy Systems. 1493-1499.(1996).

[18] Wei, Jianlin, Jihong Wang, and Shen Guo. "Mathematic modeling and condition monitoring of power station tube-ball mill systems." 2009 American Control Conference. IEEE, 2009.

[19] Zhao, Jin, and Bimal K. Bose. "Evaluation of membership functions for fuzzy logic controlled induction motor drive." IECON 02 [Industrial Electronics Society, IEEE 2002 28th Annual Conference of the]. Vol. 1. IEEE, 2002.

[20] Snejana Yordanova, Daniel Merazchiev, and Lakhmi Jain, "A Two-Variable Fuzzy Control Design With Application to an Air-Conditioning System", IEEE Transactions on fuzzy systems, Vol. 23, no. 2, (April 2015).