

DESIGNING INTELLIGENT PROCESS CONTROL SYSTEM USING FUZZY-NEURAL NETWORK

Harmony Nnenna Nwobodo

(Dept of Computer Engineering, Enugu State University of Science and Technology, Enugu)

and

Hyacinth Chibueze Inyama

(Dept of Electronic and Computer Engineering, Nnamdi Azikiwe University, Awka)

Abstract-This paper presents the design of a process control system using Fuzzy Neural Network (FNN). A Fuzzy Neural Network (FNN) or NeuroFuzzy system is a learning machine that finds the parameters of a Fuzzy System by exploiting approximation techniques from neural networks. The FNN is based on a Fuzzy System which is trained by a learning algorithm derived from Neural Network theory. A PID continuous water process control system that lets water be at a set level as water flows in and out of the tank was considered. In this paper, the hybrid of the Fuzzy Logic and Neural Network was implemented to fine tune and optimize the Fuzzy output. The output of the Fuzzy Logic alone and the output of corresponding Neural Network System were compared with the output of the hybrid: Fuzzy Neural Network. It was observed that the Intelligent Process Control Systems Using Fuzzy Neural Network gave a steep rise time of 4s and stabilized after 6s. The Intelligent Process Control Systems using Fuzzy Network rose sluggishly after an undershoot at 5s and stabilized after 9s while the Intelligent Process Control Systems using Neural Network alone had a rise time at 3s and stabilized after 9s. The Fuzzy Neural Network a hybrid of the two performed better than the Fuzzy System or the Neural Network alone, considering the rising time and stabilizing time. The Fuzzy Neural Network performed best with a rise time of 1s and stabilizing time of 9s and showed a smoother overall performance.

KEYWORDS: Neurofuzzy, fuzzy sets, Neuron, brain, connection weight, feed-forward, back-propagation, Membership Functions, Fuzzy Rule

(I) INTRODUCTION

Both Neural Networks and Fuzzy Systems have some things in common. They can be used for solving a problems (e.g. pattern recognition, regression or density estimation) if there does not exist any mathematical model of the given problem. They solely do have certain disadvantages and advantages which almost completely disappear by combining both concepts. Neural networks can only come into play if the problem is expressed by a sufficient amount of observed examples. These observations are used to train the black box. On the one hand no prior knowledge about the problem needs to be given. On the other hand, however, it is not straightforward to extract comprehensible rules from the neural network's structure. On the contrary, a fuzzy system demands linguistic rules instead of learning examples as prior knowledge. Furthermore the input and output variables have to be described linguistically. If the knowledge is incomplete, wrong or contradictory, then the fuzzy system must be tuned. Since there is no formal approach for it, the tuning is performed in a heuristic way. This is usually very time consuming and error-prone

Zhou and Quek [1] showed that Neural Networks with their remarkable ability to derive meaning from complicated or imprecise data can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an

expert in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer what if questions. Seema, Mitra and Vijay [2] designed a Neural Network Tuned Fuzzy controller for Multiple Input Multiple output (MIMO) system featuring a neural network based tuned fuzzy controller for controlling the degree of freedom for MIMO systems. The coupling effect of the system was added into the main fuzzy controller for each step to improve the performance. A data set generated was partitioned into a set of clusters based on subtractive clustering method. A fuzzy IF-THEN rule was then extracted from each cluster to form a fuzzy rule base from which the fuzzy neural network is designed. The neural network designed contains only three layers and the hybrid learning algorithm was used to refine the parameters on fuzzy rule base. Mitchell, Lopes, Fidalgo and McCalley [3] worked on using a Neural Network to predict the Dynamic Frequency Response of a power system to an Under-frequency Load Shedding Scenario. It showed a method to quickly and accurately predict the dynamic response of a power system during an under frequency load shedding. Emergency actions in a power system due to loss of generation typically calls for under-frequency load shedding measures, to avoid potential collapse due to lack of time in which to correct the imbalance via other means. This is a slow and repetitious use of dynamic simulators so a neural network was used to obtain a fast and accurate procedure during optimal load shedding. Inyama H.C and Nwobodo H.N [4] worked on Designing Intelligent Process Control System using Fuzzy Logic Principles. Inyama H.C

and Nwobodo H.N [5] also worked on Designing Intelligent Process Control System using Neural Network. In this particular paper, the architecture of Fuzzy Neural Network was discussed, followed by the design of Fuzzy Neural Network for the process control system. Finally the Fuzzy Neural Network was simulated using Proteus software. Fuzzy Neural Network when compared with Fuzzy Logic Principles and Neural Network showed the best performance out of the three techniques.

(II) Architecture of Fuzzy Neural Network

A fuzzy-neural system may be viewed as a 3-layer feed forward neural network Fig 1.1. The first layer represents the input variables, the middle (hidden) layer symbolizes the fuzzy rules and the third layer corresponds to the output variables. Fuzzy sets are encoded as (fuzzy) connection weights. Some approaches also use five layers where the fuzzy sets are encoded in the units of the second and fourth layer, respectively. However, these models can be transformed into three-layer architecture.

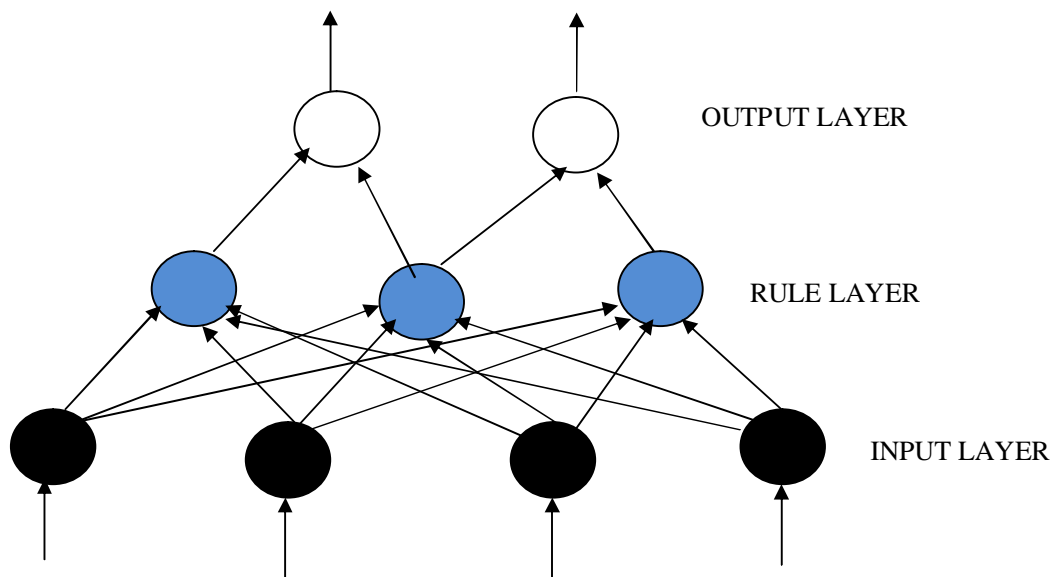


FIG 1.1 THREE LAYER FEED FORWARD

A fuzzy-neural system can always be interpreted as a system of fuzzy rules (i.e. before, during and after learning). It is also possible to create the system out of training data from scratch and it is also possible to initialize it by prior knowledge in form of fuzzy rules. The learning procedure of a Fuzzy Neural system takes the semantical properties of the underlying Fuzzy System into account. This results in constraints on the possible modifications applicable to the system parameters. A Fuzzy Neural System approximates an unknown function that is partially defined by the training data. The fuzzy rules encoded within the system represent vague samples, and can be viewed as prototypes of the training data.

Two different kinds of Fuzzy Neural Networks (FNN) were discussed in this paper: They are Cooperative FNN and Hybrid FNN [6]

i) Cooperative Fuzzy-Neural Network

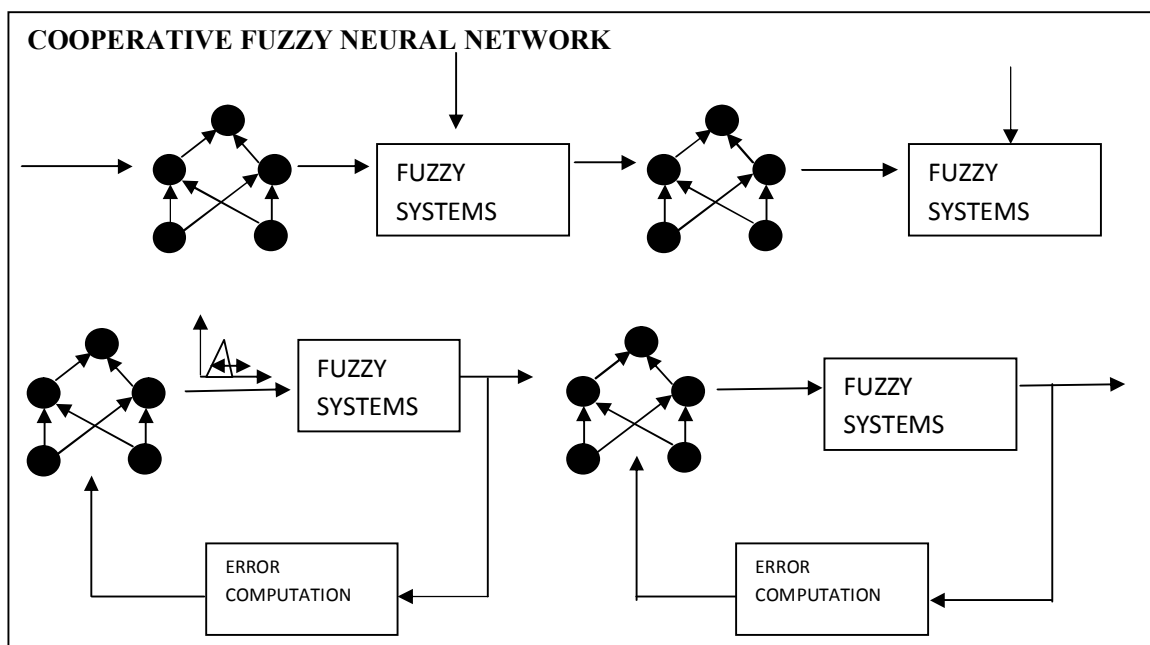


Figure 1.2 Cooperative Fuzzy Neural Networks

In Cooperative Fuzzy Neural Network, both Artificial Neural Network (ANN) and Fuzzy System work independently from each other. The ANN tries to learn the parameters from the Fuzzy System. This can either be performed offline or online while the Fuzzy System is applied. Figure 1.2 depicts four different kinds of Cooperative Fuzzy Neural Networks. The upper left fuzzy neural network learns fuzzy set from given training data. This is usually performed by fitting membership functions with a neural network. The fuzzy sets are then determined offline. They are then utilized to form the fuzzy system by fuzzy rules that are given (not learned) as well.

The upper right Fuzzy-Neural System determines fuzzy rules from training data by a neural network. Here also, the Neural Networks learn offline before the fuzzy system is initialized. The rule learning is usually done by clustering on self-organizing feature maps. It is also possible to apply fuzzy clustering methods to obtain rules.

In the lower left Fuzzy-Neural model, the system learns all membership function parameters online, i.e. while the fuzzy system is applied. Thus initially fuzzy rules and membership functions must be defined beforehand. Moreover, the error has to be measured in order to improve and guide the learning step.

The lower right Fuzzy-Neural model determines rule weights for all fuzzy rules by a neural network. This can be done online and offline. A rule weight is interpreted as the influence of a rule. They are multiplied with the rule output. The authors argue that the semantics of rule weights are not clearly defined. They could be replaced by modified membership functions. However, this could

destroy the interpretation of fuzzy sets. Moreover, identical linguistic values might be represented differently in dissimilar rules.

ii) Hybrid Fuzzy-Neural Network

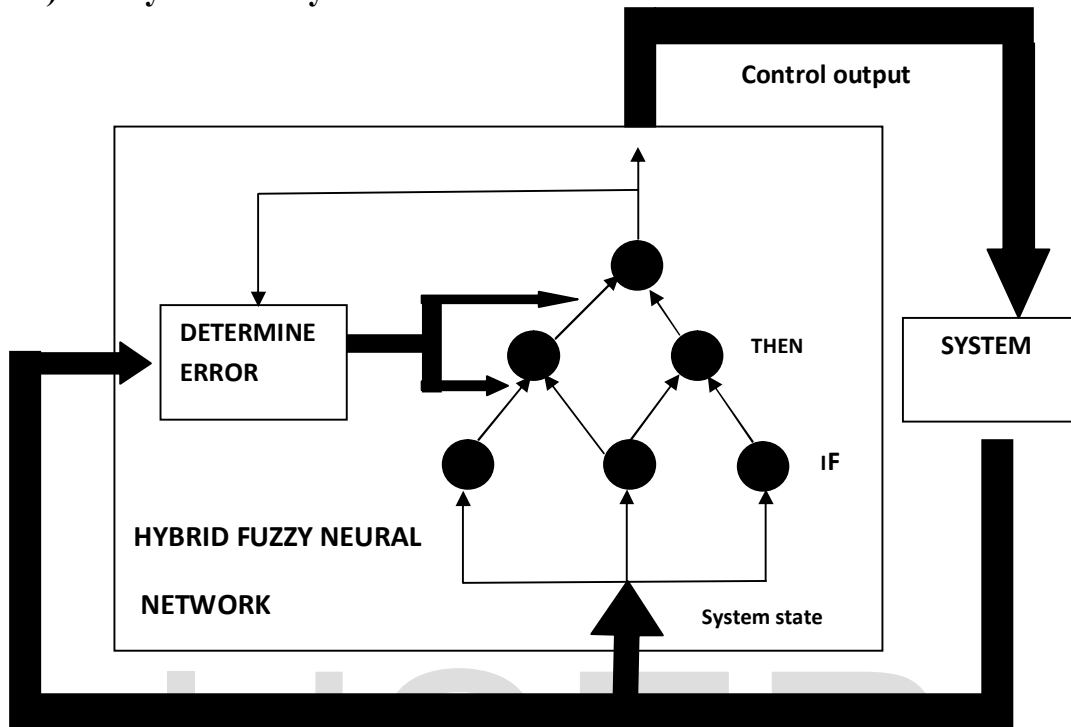


Figure 1.3 Hybrid Fuzzy Neural Networks

Hybrid Fuzzy Neural Network is homogeneous and usually resembles Neural Networks. Here, the Fuzzy System is interpreted as special kind of neural network. The advantage of such hybrid FNN is its architecture, since both Fuzzy System and Neural Network do not have to communicate any more with each other. They are one fully fused entity. These systems can learn online and offline. Figure 1.3 shows such a Hybrid FNN, here the rule base of a Fuzzy System is interpreted as a Neural Network. Fuzzy sets can be regarded as weights whereas the input and output variables and the rules are modeled as neurons. Neurons can be included or deleted in the learning step. Finally, the

neurons of the network represent the fuzzy knowledge base. Obviously, the major drawbacks of both underlying systems are thus overcome.

In order to build a fuzzy controller, membership functions which express the linguistic terms of the inference rules have to be defined. In fuzzy set theory, no formal approach to define these functions. Any shape (e.g., triangular, Gaussian) can be considered as membership function with an arbitrary set of parameters. Thus the optimization of these functions in terms of generalizing the data is very important for fuzzy systems. Neural networks can be used to solve this problem by fixing a distinct shape of the membership functions, say triangular. The neural network must optimize their parameters by gradient descent. Thus, besides the information about the shape of the membership functions, training data must be available as well.

(III) The Fuzzy Neural Network Design

Fuzzy Neural Networks (FNN) combines fuzzy logic with Artificial Neural Networks. In this work, the input and output variables processed by the fuzzy logic was fed into the neural networks for learning. (This step is called fuzzification). The neural network then takes the fuzzified input and output scales to derive a model which is converted back to the original input and output scales (defuzzification). Then the defuzzified output becomes input to the process under control. The FNN shown in Fig 1.4 has four feed forward layers. The first layer represents the input variables for liquid level, temperature and pressure respectively, the second represents their membership sets, the third layer represents the fuzzy rules and the fourth layer represents the defuzzified

output for the three variables under consideration. The rule layer produces seventy-five rules altogether, twenty-five rules for each of the variables and gives three outputs at a time, one for each of the three variables and that satisfies the output requirement at a particular instance.

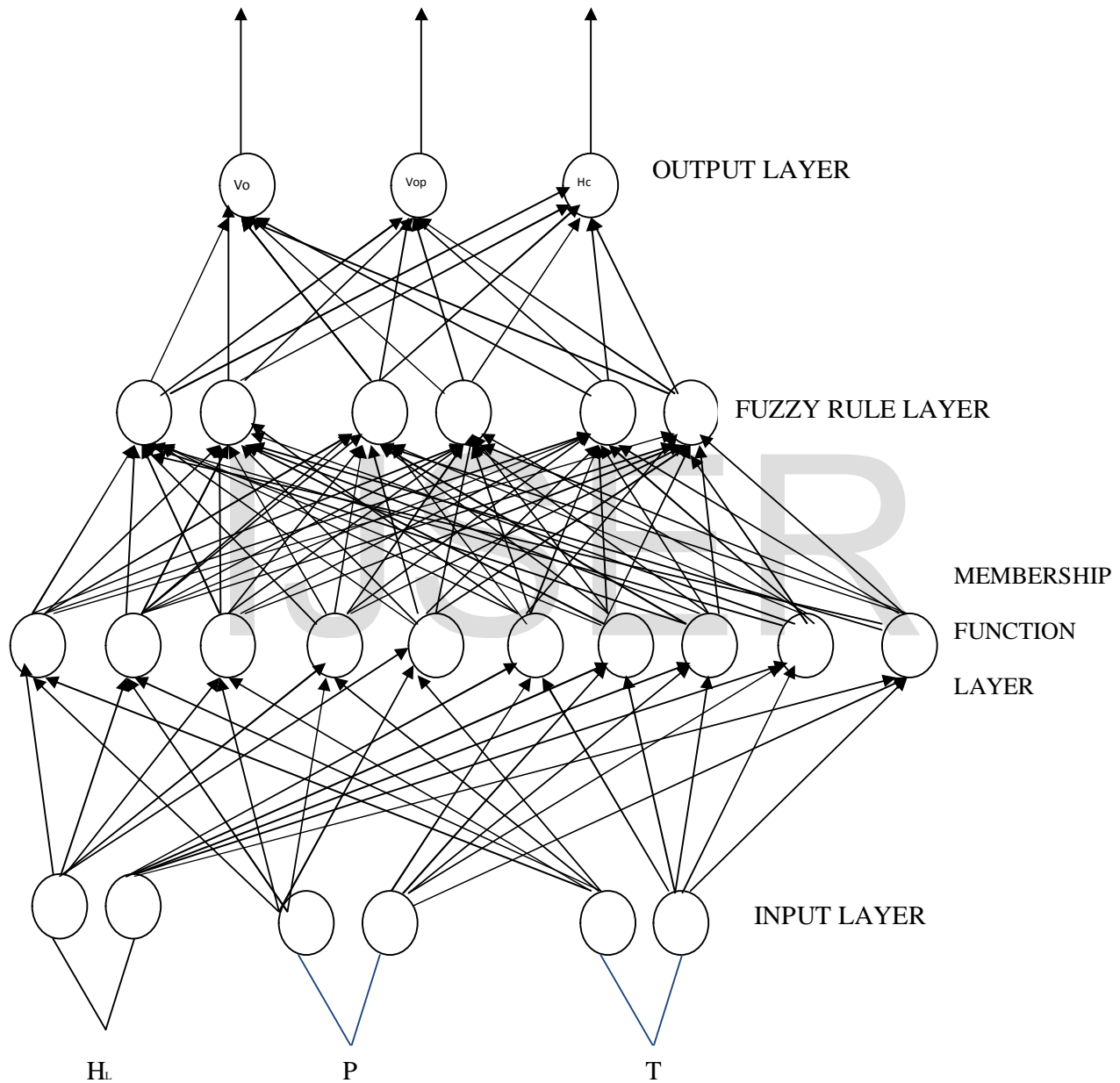


Figure 1.4 The Fuzzy Neural controller of tank water level

Fig 1.5 shows the block diagram of the Fuzzy-Neural control system running on PC that is interfaced to the process under control via an INTEL 8286

Bidirectional Buffer. Just like in the neural controller, when the signal line B is active, the buffer sends information to the process under control via the output port latch. Similarly, when the signal line A is active, the feedback signals from the process are input via the input port latch to the bidirectional buffer and from thence to the PC. The process under Fuzzy-neural control has a valve that controls the inflow of the liquid into the tank and another one that controls the outflow from the tank. The heater increases the temperature of the liquid while the stirrer is used to ensure that the liquid is of uniform temperature. The control system tries to keep the liquid level in the tank constant within the set-point for level. This is achieved by fuzzy-neural control by adjusting the inflow rate and outflow rate dynamically as appropriate. Four sensors were also used: 1) to sense the liquid level in the tank and 2) to sense the outflow rate from the tank, 3) to sense the temperature and 4) to sense the pressure of the system. The four sensor outputs are selected one at a time via 4X1 analog multiplexer. The selected signal is first amplified and then converted to digital pattern via an analog to digital converter. The digitalized sensor output are latched by the input port latch and forwarded to the fuzzy-neural control system via the bidirectional buffer.

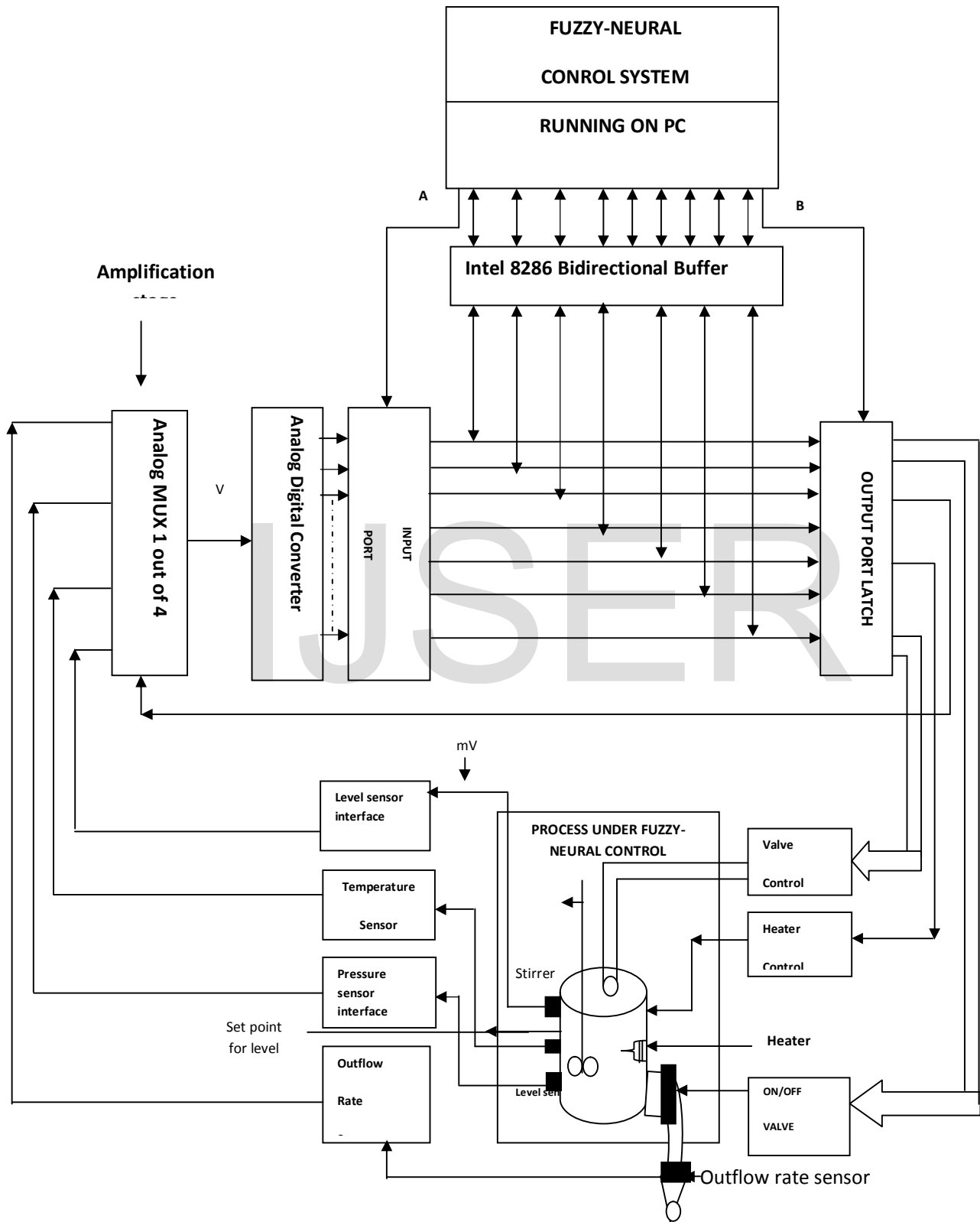


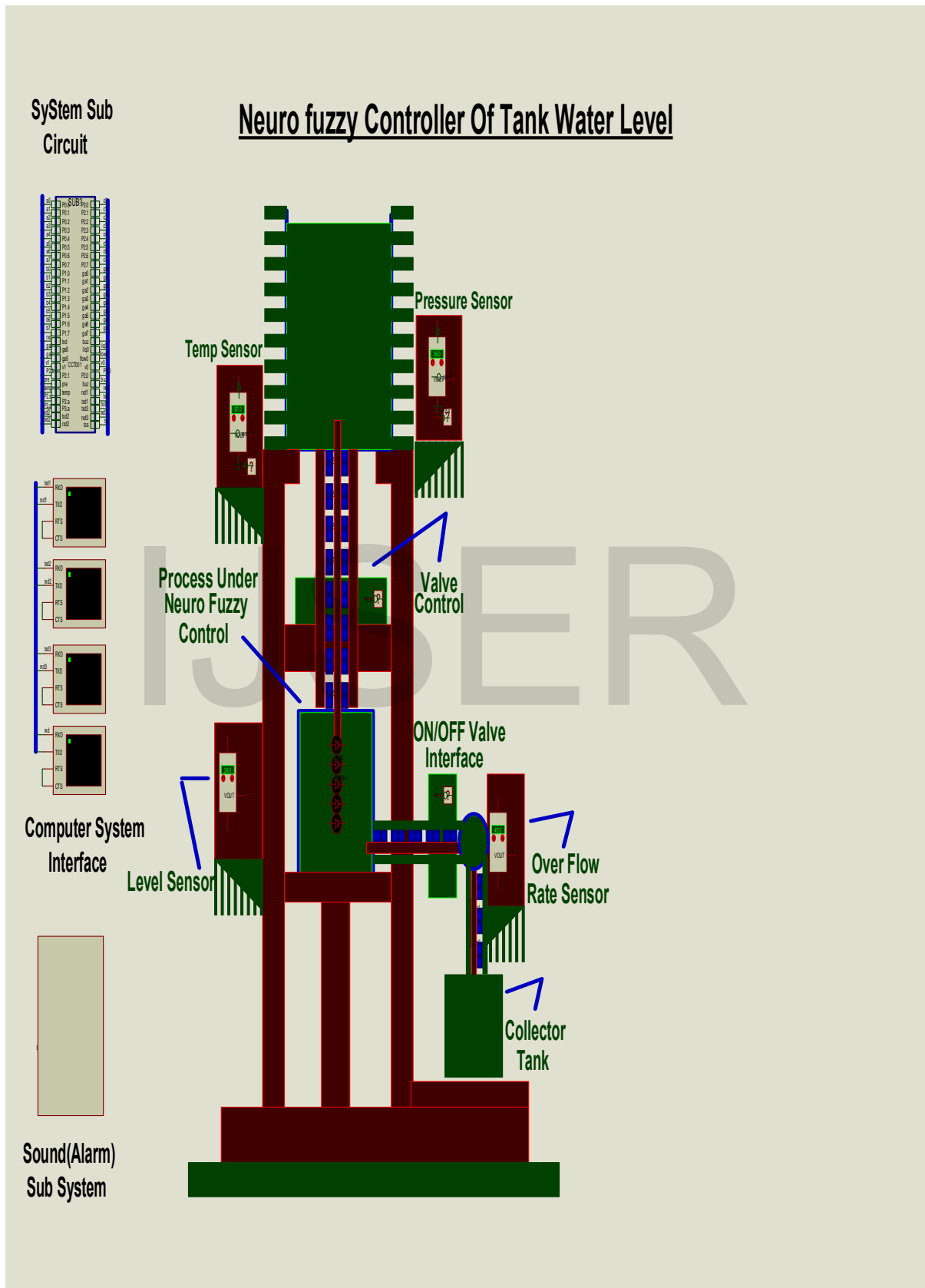
Figure 1.5 Block diagram of The Intelligent Process Control System Using Fuzzy-Neural Network

IV THE INDUSTRIAL SCENARIO

It is always necessary to perform a real-time simulation of any control system before the circuit is constructed, to ensure the workability of that system. In this work Proteus software package was used for the simulation.

The real time simulation of the control system used as a typical industrial scenario is activated by clicking the play button. When this is done the liquid from the upper tank flows to the lower tank. The valve at the water inlet of the upper tank is being controlled based on the rate of the outflow from the lower tank. This is done to make sure that the set-point of the process under control is maintained. The heater heats the liquid while the stirrer ensures that a uniform liquid temperature is maintained. Fig 1.6 shows the industrial scenario of the process control in Proteus environment. The three computer system interfaces shown display the variables under control (liquid level/outflow rate, pressure and temperature). It should be noted that when Proteus software is used to implement a real-time simulation, not all the system components are placed on the layout. Some are placed in the sub sheet for example the controller and some components are not visible but are there by default e.g. reset circuit, bidirectional buffer, power, the interfaces, analog to digital converter, the

multiplexer etc; while some are taken care of by the control program. In Proteus design, this is called *referencing*.



V. RESULT

Table 1 shows the output of the Fuzzy Logic, table 2 shows the output of the Neural Network while Table three shows the output of Fuzzy Neural Network

Table 1 data output Fuzzy Logic

R_{out} (cm³/s)	23,5	24,3	25,1	30.4	50,5	90.3	100.8	112.2	111.0	112.3	112.5	118.3	120.6	119.9
T(s)	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Table 2 data output Neural Network

R_{out} (cm³/s)	24	24	25	30	51	90	101	112	111	112	113	118	121	120
T(s)	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Table 2 data output Fuzzy Neural Network

R_{out}(cm³/s)	23	24	24	24	110	110	111	111	112	112	119	119	120	120
T(s)	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Figs 1.7, Fig 1.8 and Fig 1.9 show the graph of output of Fuzzy Logic for water level, graph of output of Neural Network and that of Fuzzy Neural Network respectively.

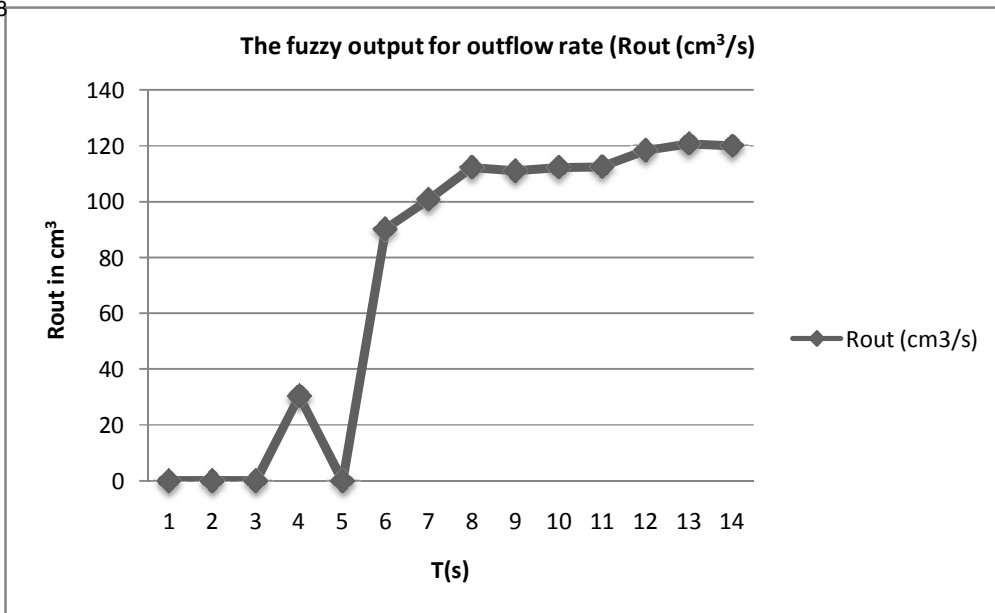


Fig 1.7 graph of Fuzzy output for PID tank water level [4]

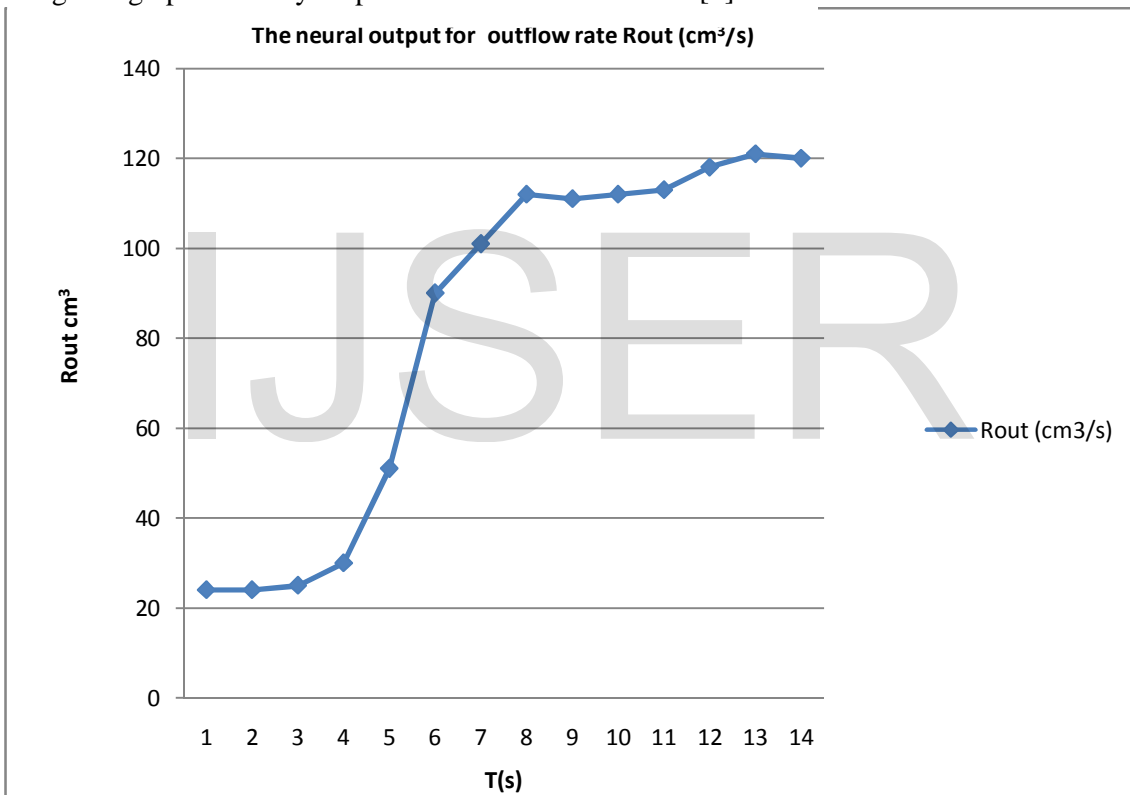


Fig 1.8 graph of Neural Network output for PID tank water level [5]

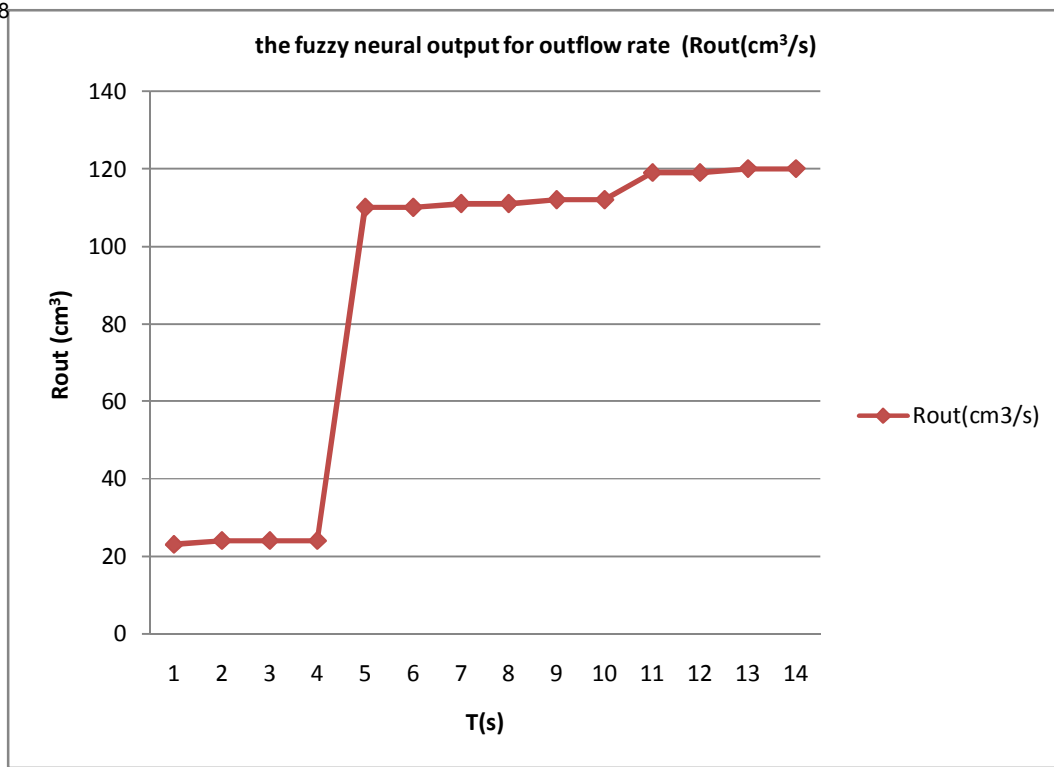


Fig 1.9 graph of Fuzzy Neural Network output for PID tank water level

IJSER

VI. CONCLUSION

Figs 1.7 through 1.9 show the graphs of the output of the three Networks. It was observed that the Intelligent Process Control Systems Using Fuzzy Neural Network gave a steep rise time of 4s and stabilized after 6s. The Intelligent Process Control Systems using Fuzzy Network rose sluggishly after an undershoot at 5s and stabilized after 9s while the Intelligent Process Control Systems using Neural Network kicked off its rise time at 3s and stabilized after 9s. The Fuzzy Neural Network performed better than either the Fuzzy System alone or the Neural Network alone, considering the rising time and stabilizing

time. The Fuzzy Neural Network performed best with a rise time of 1s and showed a smoother overall performance. The three network graphs are plotted together in Fig 1.10 to further highlight these observations.

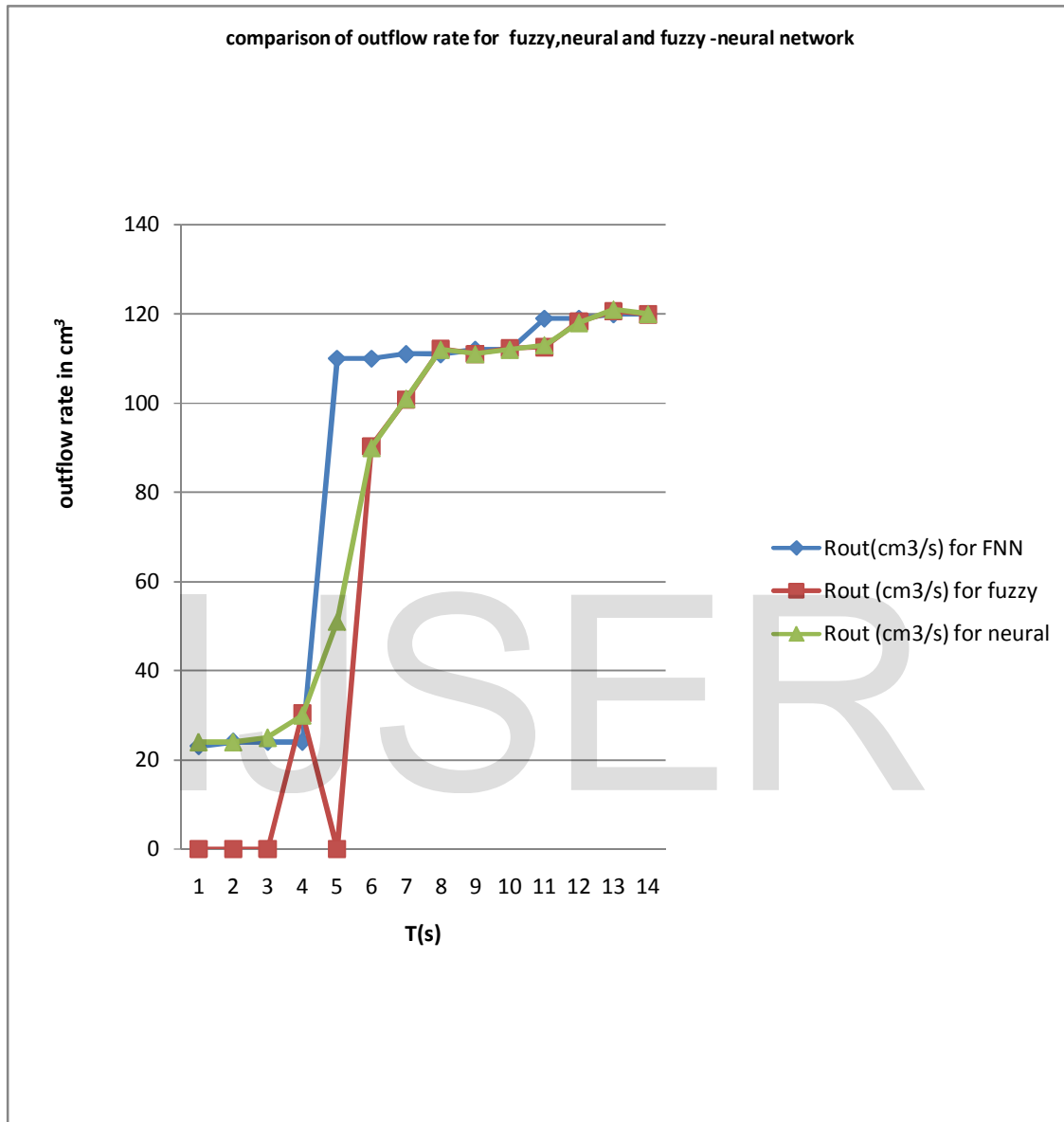


Fig 1.10 The graph of combination Fuzzy, Neural and Fuzzy-Neural Network

(VII) REFERENCES

- 1) Zhou, R. W. & Quek, C. (1996). "POPFNN: A Pseudo Outer-product Based Fuzzy Neural Network". *Neural Networks*, 9(9), 1569-1581.
- 2) Seema Chopra, Mitra R, Vijay Kumar, Neural Network Tuned fuzzy controller for Multiple Input Multiple output (MIMO), World Academy of Science, Engineering and Technology 2007.(p.485-491)
- 3) Mitchell Matthew .A, Lopes Pecas J.A, Fidalgo J.N and McCalley James D, Neural Network to predict the Dynamic Frequency Response of a power system to an Under-frequency Load Shedding Scenario, I.E.E.E Journal 2000 (p. 346-351).
- 4) Inyama H.C and Nwobodo H.N, Designing Intelligent Process Control System Using Fuzzy logic Principles, International Journal Of Scientific Research And Education, August 2014
- 5) Inyama H.C and Nwobodo H.N, Designing Intelligent Process Control System Using Neural Network, International Journal Of Scientific Research And Education, August 2014

Synergism to Intelligent Systems. Upper Saddle River, NJ: Prentice Hall.

IJSER