Computing, Information Systems, Development Informatics & Allied Research Journal
Vol. 7 No. 4, December, 2016 - www.cisdijournal.net

A Hybrid technique for Job Shop Scheduling in an Interconnected System

Onwuachu, U.C
Department of Computer Sciences,
Imo state University,
Owerri, Nigeria.
E-mail: uzochukwuchristian@yahoo.com

Enyindah, P.
Department of Computer Sciences,
University of Port Harcourt,
Choba, Nigeria.
E-mail: promise.enyindah@uniport.edu.ng

ABSTRACT

Job scheduling is an interconnected system is an important issue because it determines the performance of the system. An appropriate scheduling algorithm can efficiently reduce the response time, turnaround time and further increase the throughput. However, finding an optimal scheduling algorithm becomes a necessity. This paper proposes a system that uses neurofuzzy logic in job allocation and job sequence or a dispatching rule in an interconnected system. The proposed system is made up of two phases; the first phase uses Neural Network to determine the job priority and the job processing time base on some factor that were considered as the input variable. The second phase uses fuzzy logic techniques to determine job scheduling bases on the job processing time and the job Priority. The system was implemented using MatLab 2008. It was designed to meet up with the timing, sequencing, routing and priority setting. From the result obtained, the system was able to achieve load balancing and minimize the job processing time.

Keywords: Fuzzy logic, Neural Network, job scheduling, job processing time, job Priority and interconnected system.

CISDI Journal Reference Format
Onwuachu, U.C. & Enyindah, P. (2016): A Hybrid technique for Job Shop Scheduling in an Interconnected System. Computing, Information Systems, Development Informatics & Allied Research Journal. Vol 7 No 4. Pp 205-216
Available online at www.cisdijournal.net

1. INTRODUCTION

The resource-constrained task-scheduling problem is at the heart of many scheduling problems (Anurag et al, 2006). Real-time systems are systems in which the time at which the result is produced is as important as the logical correctness of the result (Arpana et al, 2014). Scheduling is an important aspect in real-time systems to ensure soft and hard timing constraints. Scheduling tasks involves the allotment of resources and time to jobs, to satisfy certain performance needs. In a real-time application, real-time jobs are the basic executable entities that are scheduled. (Saad et al, 2009). Scheduling a job set implies planning the order of execution of task requests so that the timing constraints are met. The schedule is said to be feasible if the timing constraints of all the tasks are met. All scheduling algorithms face the challenge of creating a feasible schedule.

The main objective of task scheduling in real-time systems is achieving high resource utilization. Scheduling is the task that determines the job allocation and job sequence at each machine. The Fuzzy scheduler considers the sequencing of job in the scheduling problem. The sequencing of jobs was approached using fuzzy controllers having rules with two antecedents which include the job processing time and the job Priority (Ziaulet al, 2012). The proposed system is made up of two phases; the first phase uses Neural Network to find out job priority based on some factors that were considered.

The second phase uses fuzzy logic techniques to determine job scheduling bases on the job processing time and the job Priority of each job waiting in a system buffer, so that whenever the system are free the job with the highest priority among those waiting is chosen. This paper proposed a fuzzy logic algorithm using fuzzy membership functions to model job scheduling problem. The solution sets will be pooled together and a final solution is obtained to minimize dissatisfaction amongst the decision makers.

2. LITERATURE REVIEW

Fatin and Safanah (Fatin & Safarah, 2015), Intelligent Neural Network with Greedy Alignment for Job-Shop Scheduling. In their paper, three layers Feed Forward Backpropagation Neural Network (FFBNN) was used for two different purposes, the first one task is to obtain the priority and the second one role is to determine the starting order of each operation within a job. The combined greedy procedure along with the back propagation algorithm will align operations of each job until best solution is obtained. However, adding a predefined alignment dataset along with the greedy procedure result in optimal solutions

Anurag et al (2006), An Improved Augmented Neural-Network Approach for Scheduling Problems, they propose an augmented neural-network approach, which allows the integration of greedy as well as non greedy heuristics (AugNN-GNG), to give improved solutions in a small number of iterations. The problem we address is that of minimizing the makespan of n tasks on m identical machines (or processors), where tasks are non preemptive and follow a precedence order. The proposed approach exploits the observation that a non greedy search heuristic often finds better solutions than do their greedy counterparts.

Forough et al (2014), Job Scheduling Problem with Fuzzy Neural Network by using the Map Reduce Model in a Cloud Environment, in their work fuzzy neural network must be use to solve this kind of optimization problem. In this paper, we proposed new novel method by using a fuzzy neural network with map reduce model to solve job shop scheduling problem, implementation and results are presented. The experiments of our proposed method are performed for well-known problem instances from job scheduling. The results show our method has high convergence speed and less execution time compared with Genetic algorithm (Forough et al, 2014).

Venkatesa & Dinesh (2014) worked on Job Scheduling Using Fuzzy Neural Network Algorithm in Cloud Environment. In their work the classified tasks are given to fuzzier where the input values are converted into the range between 0 and 1. The exemplar input is matched with the exemplar output label by adjusting weights. The algorithm is implemented with the help of simulation tool (CloudSim) and the result obtained reduces the total turnaround time and also increase the performance.
.
Rukhsana (2014) in Neural Network for Solving Job-Shop Scheduling Problem used the heuristic method which gives a high quality approximate solution in reasonable time. The simulation of the proposed method has been performed on various benchmarks. For two jobs and three machines (2/3/J/Cmax) dataset problem and any dataset problems, the simulation results shows the efficient with respect to the resolution speed, quality of the solution, and the reduction of the computation time which was not solved by Fanaiech et al.

So, the simulation results have revealed that proposed heuristic algorithm can find high quality solutions to large sized instances very quickly (Rukshana, 2014). Saxena et al (2013) worked on Load Scheduling Algorithm Prediction for Multiple Tasks using Time Series Neural Network. A time series algorithm in neural network is used for obtaining optimal schedules and the neural network is trained on these schedules. Knowledge is extracted from the trained network and past history data. The performance of this extracted rule set scheduling instances. It aims to optimize job specific parameters as well as the resource utilization. The scheduling system was able to dynamically calculation of finishing time and time consumed in scheduling the tasks of different parameters.[12]

Ziaul et al. (2012) developed a flexible manufacturing systems (FMS) using a fuzzy-multicriteria based approach for job sequencing and routing. They applied Fuzzylogic based simulation and consider number variables with reasonable amount of accuracy.There proposed model prioritize the job and select the best alternative route with multicriteria scheduling through an approach based on a fuzzy logic. There are three criteria for both the jobsequencing and routing with 27 rules. With the help of the rules the sequence of the jobs are done and the best route is selected.

Marek and Roman (2015) worked on re-planning in predictive-reactive scheduling. They looked at the techniques related to predictive-reactive scheduling and suggest the future goal, which is to propose algorithms for dealing with unexpected events using the possibility of alternative processes. Ramkumar et al (2011), worked on multi criteria job shop schedule using fuzzy logic control for multiple machines multiple jobs. They use the amalgamation of fuzzy job shop scheduling approach to find profits, customer satisfaction and to solve the problem of vagueness and uncertainty using theory of fuzzy logic based in using a membership function tosolve a fuzzy mix product selection. They also present a theoretical model to demonstrate how fuzzy decision making can support the dynamic scheduling process, enabling the conflicting priorities of multi-objectives to bemanaged effectively in polynomial time.

Nagamalleswara et al (2013) worked on modified heuristic time deviation technique for job sequencing and computation of minimum total elapsed time. From their work Job sequencing problem has become the major problem in the computer field. A finite set and each operation needs to be processed during an uninterrupted period of a given length on a given machine and our Purpose is to find a schedule, that is, anallocation of the operations to time intervals to machines that has minimal length.Thus their new modified heuristic technique called time deviation method is used to obtain the required job sequence and the minimum total elapsed time is also calculated for this sequence of jobs by the usual procedure [4].

Feng et al (2008) worked on fuzzy logic based feedback scheduler for embedded control systems. A fuzzy logic based feedback scheduling approach where used to control multiple tasks sharing one embedded CPU. Execution times of these tasks and CPU workload are uncertain and imprecise. The sampling periods of control tasks are periodically adjusted with respect to uncertain resource availability. A simple period rescaling algorithm is employed, and the available CPU resource is dynamically allocated in an intelligent fashion. The proposed approach provides runtime flexibility to quality of control (QoC) management. Preliminary simulations highlight the benefits of the fuzzy logic based feedback scheduler.

Xiangzhen et al (2010) worked onefficient dynamic task scheduling in virtualized data centers with fuzzy prediction. They proposed an efficient dynamic task scheduling scheme for virtualized data centers. A graceful fuzzy prediction method is given to model the uncertain workload and the vague availability of virtualized server nodes, by using the type-I and type-II fuzzy logic systems. Experimental results show that our algorithm can improve the total availability of the virtualized data center while providing good responsiveness performance.

Taravatsadat and Napsiah (2011) worked on application of artificial intelligent in production scheduling: a critical evaluation and comparison of key approaches,in their work Production scheduling is a part of operational research which relies on combinational optimization solved by discrete methods. Their review shows that there are only few research works which compare heuristic techniques on scheduling problem. There is a need for scholars to focus on evolutionary manufacturing systems, and hybrid models to face scheduling problem. Rina and Harkut (2014), reviewed adaptive neuro fuzzy scheduler for real time task.Their work presents a review on scheduling algorithm of real time task. Then, discuss the limitations of EDF algorithm and features of neuro fuzzy system. They proposed neuro fuzzy scheduler using EDF to overcome the drawbacks of other algorithms for better scheduler performance of real time task [8].

3. MATERIALS AND METHOD

The system requirement includes the system input variables listed as

maximum completion time,
maximum makespan,
maximum lateness,
maximum cost,
maximum earliness,
maximum tardiness,
total weighted completion time,
total weighted flow time,
total weighted earliness,
total weighted tardiness,
weighted number of late jobs,
total cost.

We have a total of 12 input variables, these inputs were normalized which is an appropriate stage in training the data obtained using neural networks applications that was developed. The input data is normalized into the range of [0, 1] or [-1, 1] according to the activation function of the neurons. In this paper the value is normalized into the range of [0, 1] using a sigmoid function and the neural networks are trained and tested using the back propagation algorithm.

A. The Architectural Model of the neural network System.

The architecture of this model consists of a
12-*j*-2 network topology because of twelve input variables in which variable *j* was determined by the number of hidden neurons during network selection. Figure: 1 depicts a schematic diagram of a 12-4-2 topology and twelve input variable are: X11, X12 X13,...X112. This is the combination of both technical and fundamental variables. The hidden layer of X21, X22, X23 and X24 a intermediate variables which interact by means of weight matrices with adjustable weights to produce the output Y_1 and Y_2.

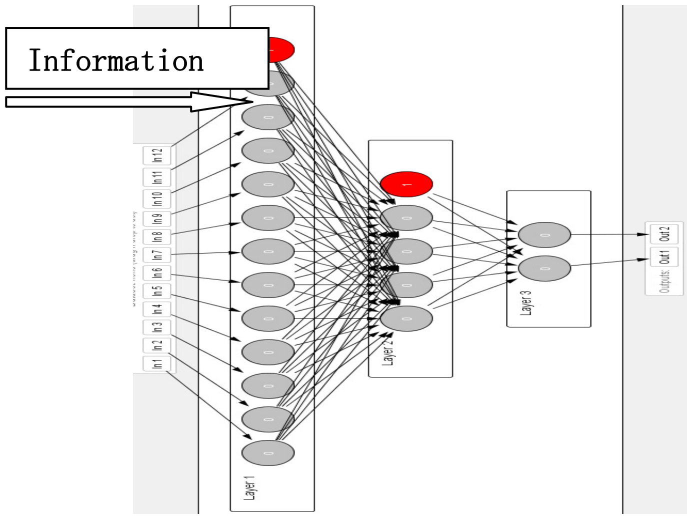

Fig: 1: Neural Network Model Architecture of the System

Forward propagation is a supervised learning algorithm and describes the "flow of information" through a neural net from its input layer to its output layer. The feed forward algorithm was used to calculate the optimal weights of the stock prediction. The mathematical models for the feed forward algorithm are as follows:

Inputj = **x**j =å **yi wi j** 1

Yi is the generated output and Wi j represents weights
$f(x) = 1 \ / \ (1 + e^{-x}{}_{j})$ ……………………2

f (x) is a sigmoid that is used as the activation function
Error $= T_k - O_k$ ……………………3

Tk is the observed (True) output while Ok is the calculated (actual) output

The error in the output layer is calculated by using the formula in equation 4
d $k = o\ k$ ***(1 - o$_k$)*(** *Tk - O k*)..... 4
Where Ok is the calculated (actual) output expressed in equation 5
O K = $1 \ / \ (1 + e^{-x}{}_{j})$ ……… …………5

Tk is the observed (True) output

The back propagation error in the hidden layer is calculated by using the formula in equation 6
d $_J = o_J$ ***(1 - o$_J$)*** å d $_K * w_{JK.}$ 6
k
Where wjk is the weight of the connection from unit j to unit k in the next layer and δk is the error of unit k.
The weight adjustment formula in equation (7) is used to adjust the weights to produce new weights which are fed back into the input layer.
W new = W old +η* δ* input 7

Where η is a constant called the learning rate. The learning rate takes value between 0 and 1.

3.1 The Basic Structure of a Fuzzy System.

Figure 2 shows the basic structure of a fuzzy expert system. It has four key modules namely, fuzzification module, fuzzy inference engine, fuzzy rule-base and defuzzification module. The fuzzification module is traditionally responsible for receiving crisp numeric measurements from the environment as input, process them and map them into fuzzy membership function values. The fuzzy engine is responsible for processing all calculated membership function values using fuzzy sets' calculations and communicates with fuzzy rule base to identify the most suitable fuzzy output. However, the defuzzification module is responsible for converting the fuzzy output into a numeric output suitable for the environment decision and control situation.

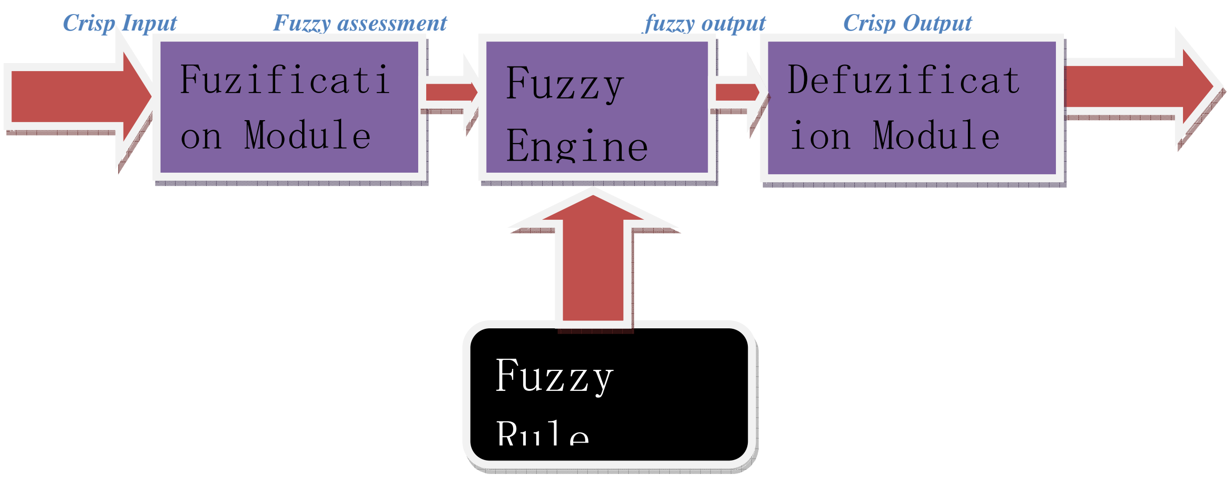

Figure 2: Basic Architecture of a Fuzzy Expert System

3.2 The Proposed System

The proposed system schedules the job in the queue based on the sequence of operations and the set of machines to be used to accomplish the operation. The fuzzy inference system calculates the membership function for the job priority and the job processing time which will be used in job allocation to different systems. The job priority is based on the importance of a job and based on the fuzzy values. The fuzzy sets the job priority JP which contains the five different fuzzy values. The customer priority fuzzy values are given as Bad, Low, Medium, High and Very Important Figure 3 shows the flow chart the proposed system. *The job Processing Time is* depending upon the processing time assigned to one particular job three fuzzy functions were assigned, for the fuzzy sets such as Short, Medium, and Long.

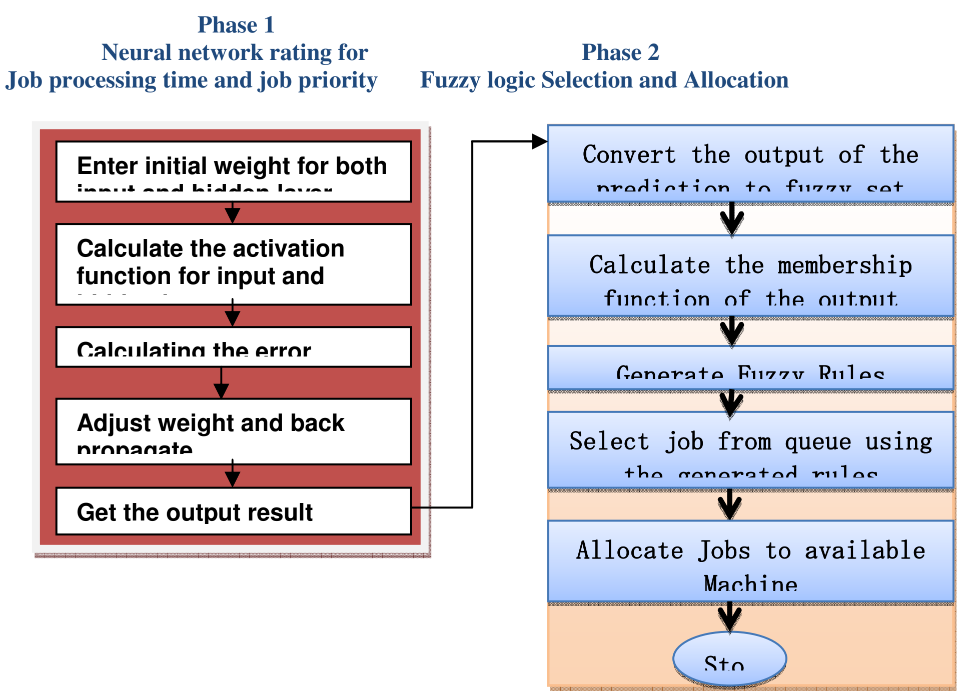

Figure. 3: A diagrammatic representation of the proposed systems framework.

4. EXPERIMENT AND RESULT

The simulation was done using Matlab 2008. Figure 4 shows the Neural Network fitting tool for data selection. This Interface helps in collection of input data and the target data from the work space.

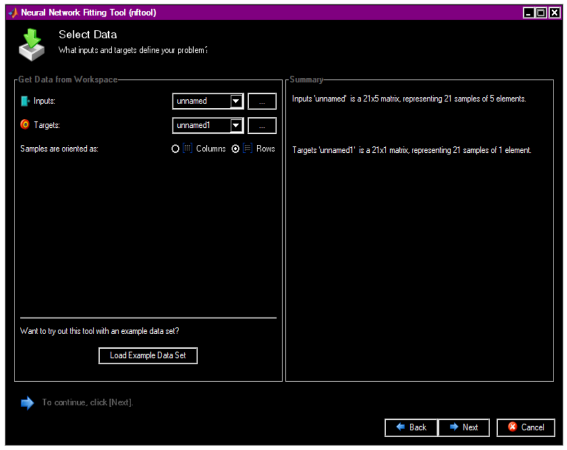

Figure 4. Neural Network Fitting Tool for Data Selection

Figure 5. shows the Neural Network fitting tool for selection of network size. This interface gives the user the opportunity to select the number of neuron in the network's hidden layer. The user can return to this panel and change the number of the neuron if the network does not perform well after training.

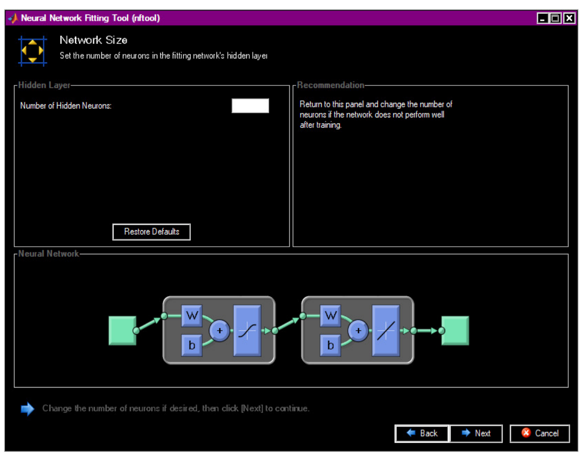

Figure 5. Neural Network Fitting Tool for Network Size Selection

Figure 6.shows the Neural Network training. The Neural Network model was trained using Levenberg-Marquardt back propagation. The network is trained to fit the inputs and the target. This means that neural network map between a data of numeric inputs and a set of numeric targets. Training automatically stops when generalization stops improving as indicated by the increase in the mean square error of the validation samples.

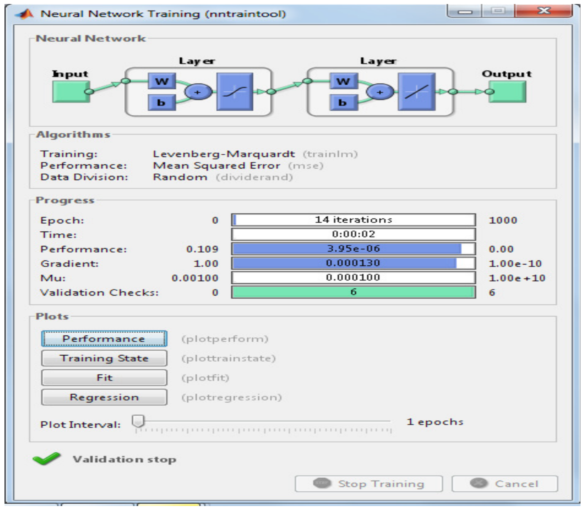

Figure 6. Neural Network Training Tool

The neural network fitting tool will help in training network and evaluation its performance using mean square error and regression analysis. Training multiple times will generate different results due to different initial condition and sampling. Figure 7. shows the result of the trained Network.

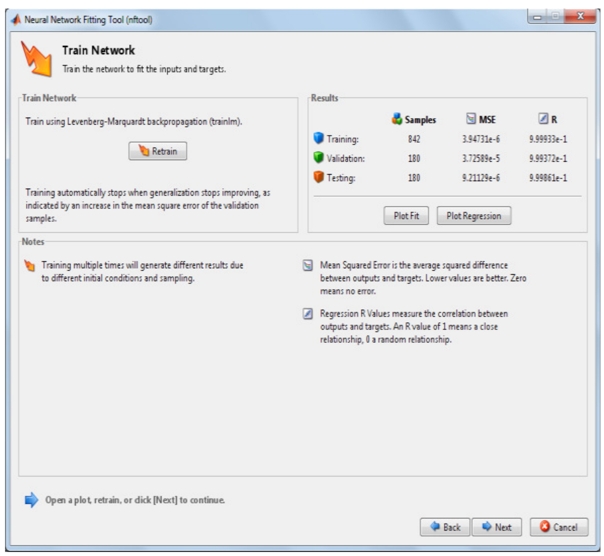

Fig ure 7. Neural Network Fitting Tool for Displaying the Result of the Trained Network

The FIS Editor displays general information about a fuzzy inference system. It has a simple diagram at the top that shows the input variables on the left, and those of each output variable on the right. The sample membership functions shown in the boxes are just icons and do not depict the actual shapes of the membership functions. Figure 8 shows the fuzzy inference system editor for job scheduling.

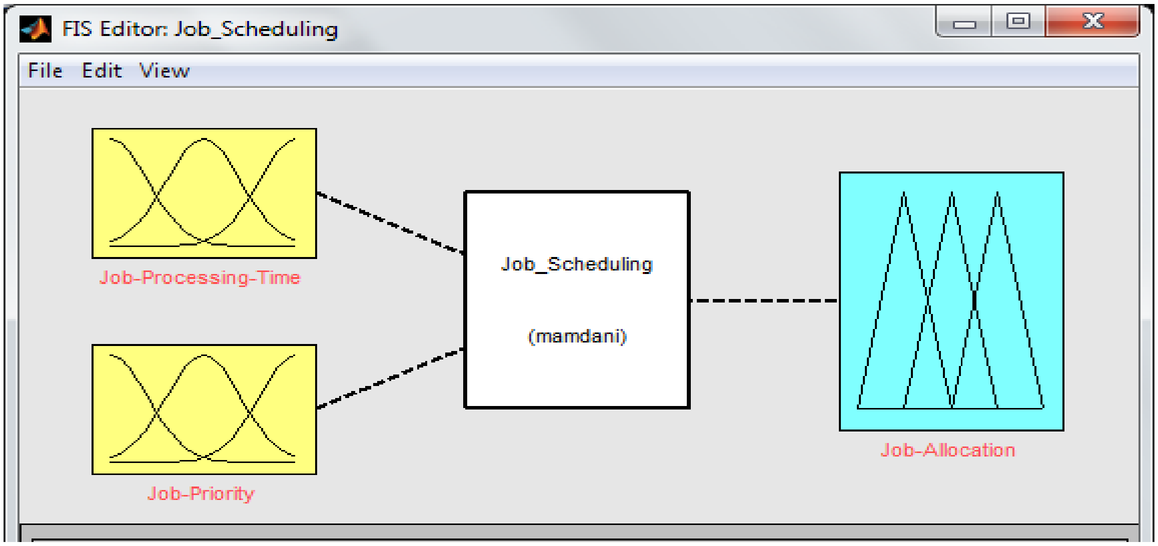

Figure 8: The fuzzy inference system editor for job scheduling.

The Membership Function Editor is the tool that lets you display and edits all of the membership functions associated with all of the input and output variables for the entire fuzzy inference system. The Membership Function Editor shares some features with the FIS Editor, as shown in the figure 4. In fact, all of the five basic GUI tools have similar menu options, status lines, and **Help** and **Close** buttons. From figure 9 we can see that the membership function for job allocation is displayed.

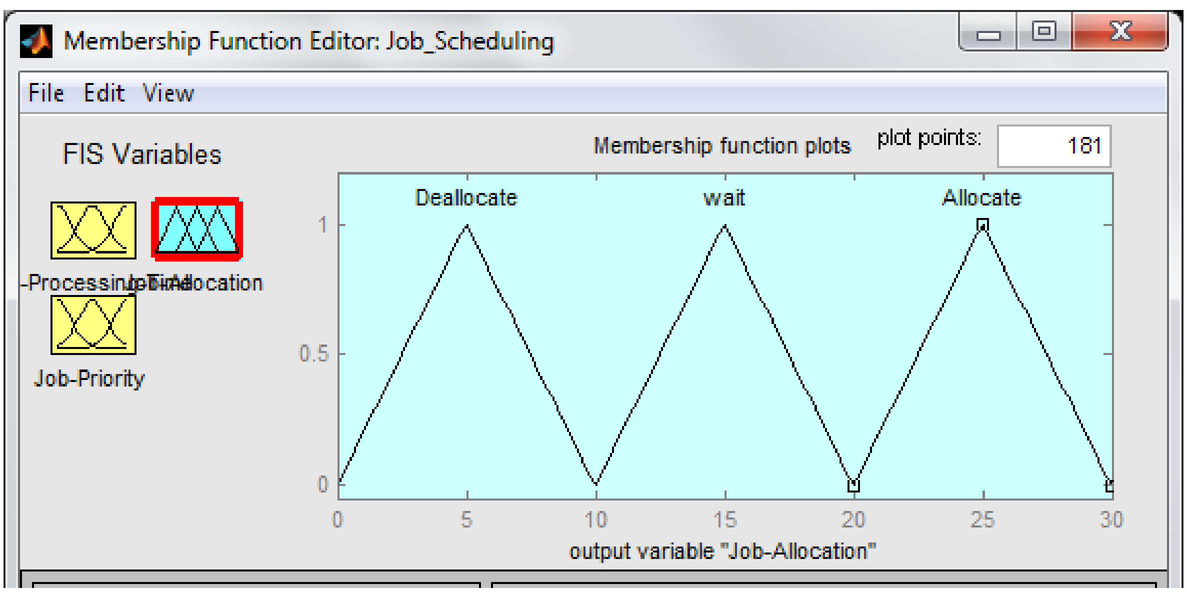

Figure 9: The fuzzy inference membership function editor for job scheduling.

Constructing rules using the graphical Rule Editor interface is fairly self-evident. Based on the descriptions of the input and output variables defined with the FIS Editor, the Rule Editor allows you to construct the rule statements automatically from the GUI. Figure 10 shows the fuzzy inference rule editor for job scheduling.

Computing, Information Systems, Development Informatics & Allied Research Journal
Vol. 7 No. 4, December, 2016 - www.cisdijournal.net

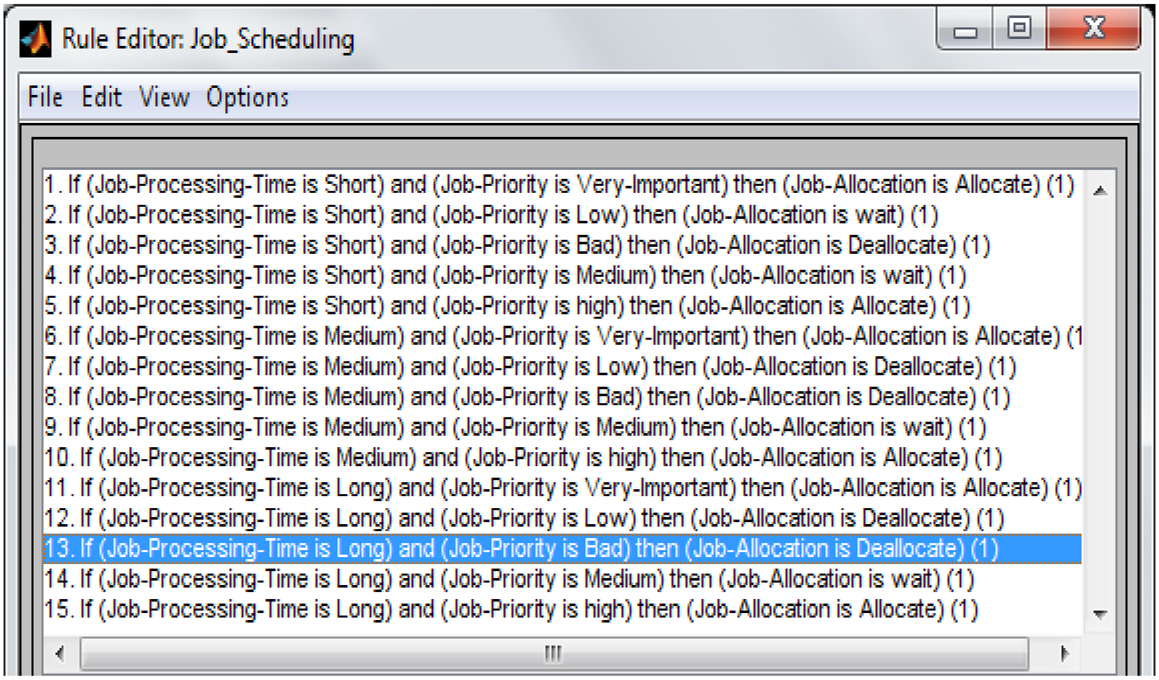

Figure 10: the fuzzy inference system rule editor for job scheduling.

At this point, the fuzzy inference system has been completely defined, in that the variables, membership functions, and the rules necessary to calculate tips are in place. The Rule Viewer displays a roadmap of the whole fuzzy inference process. The Rule Viewer allows you to interpret the entire fuzzy inference process at once. It shows how the shape of certain membership functions influences the overall result. The Rule Viewer shows one calculation at a time and in great detail. In this sense, it presents a sort of micro view of the fuzzy inference system. Figure 11 shows the fuzzy inference rule viewer for job scheduling.

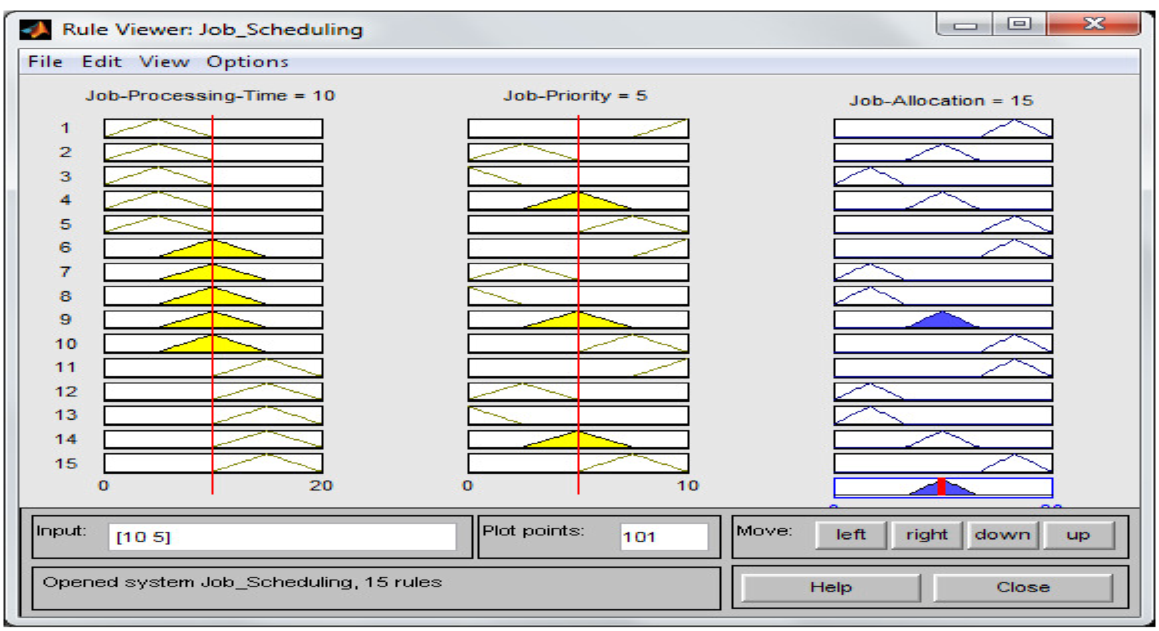

Figure 11: the fuzzy inference rule viewer for job scheduling.

The Surface Viewer has three-dimensional curve that represents the mapping from job priority and job processing time to job allocation. Because this curve represents a two-input one-output case, you can see the entire mapping in one plot. Accordingly, the Surface Viewer is equipped with drop-down menus X (input):,Y (input): and Z (output): that let you select any two inputs and any one output for plotting. Below these menus are two input fields X grids: and Y grids: that let you specify how many x-axis and y-axis grid lines you want to include. Figure 12 shows the fuzzy inference surface viewer for job scheduling.

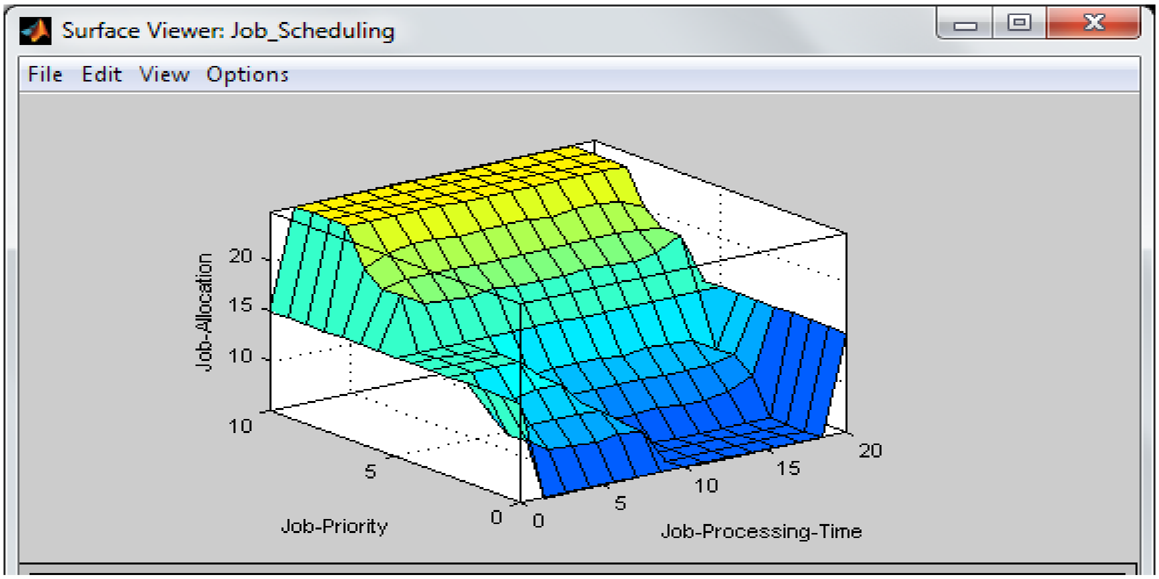

Figure 12: the fuzzy inference surface viewer for job scheduling.

5. RESULT DISCUSSION

Figure 4 shows the Neural Network fitting tool for data selection. This Interface helps in collection of input data and the target data from the work space. Figure 5. shows the Neural Network fitting tool for selection of network size. This interface gives the user the opportunity to select the number of neuron in the network's hidden layer. Figure 6.shows the Neural Network training. The Neural Network model was trained using Levenberg-Marquardt back propagation.

The network is trained to fit the inputs and the target. Training automatically stops when generalization stops improving as indicated by the increase in the mean square error of the validation samples. The neural network fitting tool will help in training network and evaluation its performance using mean square error and regression analysis. Training multiple times will generate different results due to different initial condition and sampling. Figure 7. shows the result of the trained Network. Figure 8 shows the fuzzy inference system editor for job scheduling. The FIS Editor displays general information about a fuzzy inference system. It used to edit the input and the output of the fuzzy inference system. The Membership Function Editor is the tool that lets you display and edits all of the membership functions associated with all of the input and output variables for the entire fuzzy inference system. From figure 9 we can see how the membership function for job allocation is displayed.

Figure 10 shows the fuzzy inference rule editor for job scheduling. Constructing rules using the graphical Rule Editor interface is fairly self-evident. Based on the descriptions of the input and output variables defined with the FIS Editor, the Rule Editor allows you to construct the rule statements automatically from the GUI. The Rule Viewer displays a roadmap of the whole fuzzy inference process. The Rule Viewer allows you to interpret the entire fuzzy inference process at once. It shows how the shape of certain membership functions influences the overall result as it is shown in figure 11.The Surface Viewer has three-dimensional curve that represents the mapping from job priority and job processing time to job allocation. Because this curve represents a two-input one-output case, you can see the entire mapping in one plot. Figure 12 shows the fuzzy inference surface viewer for job scheduling

6. CONCLUSION

The paper has proven that neurofuzzy logic system can be used to achieve efficacy and load balancing in job allocation and job sequence within an interconnected systems. The Scheduling algorithms are considered one of the key components of a real-time system.A fuzzy scheduling algorithm builds into the real-time system flexibility and adaptation to the uncertainty inherent in real-time environments and offers a means to improve several important characteristics of real-time systems.

REFERENCES

[1] Ziaul Hassan, Nabila Chowdhury, Abdullah-Al-MamunMasud (2012), A Fuzzy-Multicritaria Based Approach for Job Sequencing and Routing In Flexible Manufacturing System (Fms), Global Journal of Researches in Engineering Mechanical and Mechanics Engineering, Volume 12 Issue 5 Version 1.0, Online ISSN: 2249-4596 Print ISSN:0975-5861

[2] Marek Vlk and Roman Barta´k (2015), Replanning in Predictive-reactive Scheduling, Association for the Advancement of ArtificialIntelligence (www.aaai.org).

[3] R. Ramkumar, Dr. A. Tamilarasi and Dr. T. Devi (2011), Multi Criteria Job Shop Schedule Using Fuzzy Logic Control for Multiple Machines Multiple Jobs, International Journal of Computer Theory and Engineering, Vol. 3, No. 2, April 2011 ISSN: 1793-8201

[4] Nagamalleswara Rao, Dr. O. Naga Raju and Prof. I. Ramesh Babu (2013), modified heuristic time deviation technique for job sequencing and computation of minimum total elapsed time,International Journal of Computer Science & Information Technology (IJCSIT) Vol 5, No 3,

[5] Feng Xia, Xingfa Shen, Liping Liu, Zhi Wang, and Youxian Sun (2008), Fuzzy Logic Based Feedback Scheduler for Embedded Control Systems, National Laboratory of Industrial Control Technology

[6] Xiangzhen Kong , Chuang Lin , Yixin Jiang , Wei Yan , Xiaowen Chu (2010), Efficient dynamic task scheduling in virtualized data centers with fuzzy prediction, Journal of Network and Computer Applications, www.elsevier.com/locate/jnca.

[7] TaravatsadatNehzati and Napsiah Ismail (2011), Application of Artificial Intelligent in Production Scheduling: a critical evaluation and comparison of key approaches, Proceedings of the 2011 International Conference on Industrial Engineering and Operations ManagementKuala Lumpur, Malaysia.

[8] Rina V. Bhuyar and Harkut D. G. (2014), Adaptive Neuro Fuzzy Scheduler for Real Time Task: A Review, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 2, ISSN: 2277 128X

[9] Paolo Dadone (1997), Fuzzy Control of Flexible Manufacturing Systems, Thesis submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering.

[10] Saad E. M., Keshk H. A., Saleh M. A., and Hamam.A.A,(2009), Scheduling Real-Time Tasks In Multiprocessor Systems Using Genetic Algorithms, Journal of Engineering Sciences, Assiut University, Vol. 37, No. 3, pp. 691-698.

[11] Aparna Vishwanath1, Ramesh Vulavala, Sapna U. Prabhu (2014), Task Scheduling in Homogeneous Multiprocessor Systems Using Evolutionary Techniques, International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Volume 4, Issue 2,

[12] Saxena Sachin Kumar, Kumar Dhaneshwar, Singh Neha , Avasthi Vikas and Agarwal Gaurav ,(2013), Load Scheduling Algorithm Prediction for Multiple Tasks using Time Series Neural Network , International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 5, ISSN: 2277 128X

[13] Rukhsana G. Sache *(2014),* Neural Network for Solving Job-Shop Scheduling Problem, *IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 6, Ver. VIII, PP 18-25*

[14] Venkatesa Kumar V. and Dinesh K. (2012),Job Scheduling Using Fuzzy Neural Network Algorithm in Cloud Environment Bonfring International Journal of Man Machine Interface, Vol. 2, No. 1,

[15] Fatin I. Telchy and Safanah Rafaat (2015), Intelligent Neural Network with Greedy Alignment for Job-Shop Scheduling, Iraqi Journal of Computers, Communication Control and System Engineer(IJCCCE) Vol.15, No.3,

[16] Anurag Agarwal, Varghese S. Jacob and Hasan Pirkul (2006), An Improved Augmented Neural-Network Approach for Scheduling Problems, *INFORMS Journal on Computing,* Vol. 18, No. 1, pp. 119–128, issn 1091-9856 _ eissn 1526-5528 _ 06 _ 1801 _ 0119

[17] Zhang, L.;Wong, T.; Zhang, S.; andWan, S. 2011. A multiagent system architecture for integrated process planning and scheduling with meta-heuristics. In Proceedings of the 41st International Conference on Computers & Industrial Engineering.

[18] Forough zare, Mashallah Abbasi Dezfoli and Reza Javidan (2014), Job Scheduling Problem with Fuzzy Neural Network by using the Map Reduce Model in a Cloud Environment, *International Journal of Computer Applications (0975 – 8887) Volume 88 – No.14,*