Proceedings of the 7th WSEAS International Conference on Neural Networks, Cavtat, Croatia, June 12-14, 2006 (pp34-40)

Implementation of a Choquet Fuzzy Integral Based Controller on a Real Time System

SMRITI SRIVASTAVA
Department Of Instrumentation and Control Engineering
University of Delhi
Sector-3 Dwarka, New Delhi-110042
INDIA

ANKUR BANSAL
Department Of Instrumentation and Control Engineering
University of Delhi
Sector-3 Dwarka, New Delhi-110042
INDIA

DEEPAK CHOPRA
Department Of Instrumentation and Control Engineering
University of Delhi
Sector-3 Dwarka, New Delhi-110042
INDIA

GAURAV GOEL
Department Of Instrumentation and Control Engineering
University of Delhi
Sector-3 Dwarka, New Delhi-110042
INDIA

Abstract

The paper discusses about the Choquet Fuzzy Integral (CFI) based identification and control of a Real Time Dynamical System. The plant is modeled using Choquet fuzzy Integral based Neural Network with Gradient Descent as learning algorithm. This model is used for designing and simulating CFI based controller. Then this designed controller is implemented on real time system through MATLAB. Finally the performance of CFI based controller is compared with Feed Forward Neural Network taking Square of Error as the parameter of comparison.

Keywords

Real time dynamical system, Choquet Fuzzy Integral (CFI) Square of Error (SE), Gradient Descent (GD), Feed Forward Neural Network (FFNN)

1. Introduction

Modern Control theory has made tremendous success in areas where the systems are well defined, but it has failed to cope with practicalities of many industrial processes and systems. The fundamental reason for this is the lack of detailed structural knowledge of the processes and systems. To cope up with the complexity of dynamical systems, there have been significant developments in modeling and control during the last two and half decades [1][2][3]. Attempts are being made to incorporate new paradigm- Fuzzy Logic (FL), Artificial neural Networks (ANN) and Wavelet Neural Networks

(WNN) to handle uncertainty and impreciseness of the real world systems. But all these work on an assumption that most of the systems are additive but contrary to this real life time systems are non-additive.

Choquet Integral based networks does a non-linear aggregation of the inputs sets. An extensive application of fuzzy integrals to image processing can be found in [4] and to handwriting recognition in [5]. The literature is replete with applications of Choquet Integral but so far no work is reported in the field of Identification and Control. In this paper, we have used Choquet Integral as a neuro-computation for identification and control. The structure to compute the Choquet Integral is given in [5].This computation is flexible and its parameters are learned through training. This structure is transparent in nature, i.e. after training, the output node of the network is analyzed as a sub-decision and a network itself is considered as a collection of many sub-decisions. Since the structure of a Feed Forward Neural Network is analogous to that of Choquet Integral, however the method of computing the output is different, the performance of the two can be compared.

Dynamic system modeling consists of determining the structure of a model, which in turn requires a priori knowledge about the system or the input output data. In our case a priori knowledge is not known but only the input-output data is available. Using this data set, the plant is modeled using CFI. Next a CFI based controller is designed and implemented on the pressure feedback system through data acquisition cards and MATLAB. Lastly the performance of newly designed CFI controller and a FFNN controller is compared because of the obvious analogy between the two.

Our real time system consists of a plant maintaining the pressure inside a tank. The plant consists of two tanks. Tank 1 act as a source of compressed air, and Tank 2 is the one, where the pressure is to be maintained at some desired value. A pneumatic control valve maintains the flow of air between the two tanks. Some leakage is also provided in the tank 2, which acts as a disturbance. The pressure transmitter records the output pressure and transmits it to the controller. The controller sends an output voltage between 0 to 5 volts to the PCL-726 data acquisition card, which gives values between 4 to 20mA to the E/P (Electro to pneumatic) converter installed adjacent to the control valve. This converter converts 4 to 20mA signal to 3 to 15 psi, according to which the control valve opens or closes and keeps the pressure in tank 2 at the desired value.

The paper is organized as follows: Section 2 gives the overview of Choquet Fuzzy Integral along with the identification of the system using the CFI. Section 3 gives the learning of the parameters of the system by GD algorithm. Section 4 explains the design of CFI controller using the model. Interfacing between the hardware and the software i.e. how the CFI controller sends and receives the output value to and from the plant is discussed in section 5. Plant Response using CFI controller is shown in section 6. Section 7 encapsulates the discussions over the results obtained. Finally, section 8 gives conclusion about the performance of the controller

2. Modeling Real Time System using Input-output through CFI

Choquet Fuzzy Integral (CFI):

The Choquet Fuzzy integral is a fuzzy integral based on any fuzzy measure that provides alternative non-linear computational scheme for aggregating input information unlike other fuzzy and NN models. The calculation of the CFI with respect to λ fuzzy measure requires the knowledge of the fuzzy density g and the input value. CFI network is a directed graph consisting of neural nodes with interconnecting linear synaptic links and a fuzzy integral function with respect to certain fuzzy measure. The synaptic links of a neuron (called fuzzy densities) is interpreted as the degree of importance of the respective input signal. The weighted computation of the input signals defines the activity level of the neuron, which is the output value. CFI can also pick up one optimal solution if more than one exists and can increase the reality and precision of predictions and decisions in many real life problems. Every Fuzzy Integral based neuron in each layer of the network is connected to every other neuron in the adjacent layer resulting in fully connected Fuzzy Integral based Neural Network Implementation.

Consider a single layer fuzzy integral based neural network and assume M inputs h (x_1)…h (x_m) to an output node. The training data for this node is taken from M inputs sources x_1, x2… x_m with M corresponding desired outputs y_d. The learning process is to determine the best set of fuzzy densities for this node in such a way that the discrepancy between the desired and actual fuzzy integral behavior is minimized. Fig.1 shows the fuzzy integral based network.

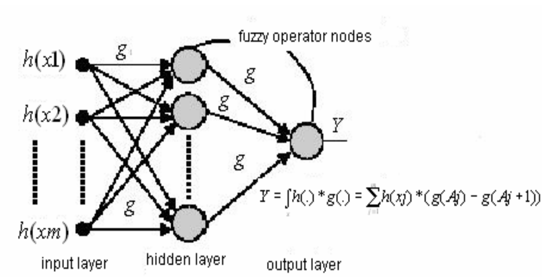

Fig.1 A fuzzy integral based network

Mathematically CFI can be expressed as

$$y = \sum_{j=1}^{m} h(x_j) * (g(A_j) - g(A_{j+1})) \qquad (1)$$

where m is the number of inputs, $g(A_j)$ is the fuzzy measure given by

$$g(Aj) = g_j + g(A_{j+1}) + \lambda g_j g(A_{j+1}) \qquad (2)$$

λ is the fuzzy measure and g_j is the fuzzy density. Hence eqn. (1) becomes

$$y = \sum_{j=1}^{m} h(x_j) * g_j(1 + \lambda g(A_{j+1})) \qquad (3)$$

For modeling the system, the main objective to use a CFI based network is to determine an adaptive algorithm or rule which adjusts the parameters of the network based on a given set of input-output pairs. The input to the system is the voltage sent to E/P converter which converts voltage into corresponding pressure and controls the opening of control valve. The output is the pressure in the tank 2 which has to be maintained close to set point.

To take the output, we send voltage to E/P converter using DAQ toolbox in MATLAB through PCL-726 card. The output is measured inside the tank using the pressure transmitter and ADAM – 4014D. The different cards are explained in section V. Outputs are measured for different input values. Densities of CFI based Network model are trained using GD algorithm.

3. Learning Algorithm

For fine tuning of densities, we chose an objective function defined as

$$J = \frac{1}{2M} \sum_{j=1}^{m} e^{2}(j) \qquad (4)$$

where $e(j) = y_d(j) - y(j)$, y_d is the desired output, y is the actual output, M is the number of data samples and j (1 to m) is the number of inputs. As J reduces, the approximation of the system is high and the densities are finely tuned. At this stage the densities are frozen so that a finely tuned system is obtained.

Parameter update formula

$$g^{new} = g^{old} + \Delta g \qquad (5)$$

where g is the density to be learned. Δg is the gradient of the density updated by the objective function. The different gradients are calculated as follows:

$$\Delta g1 = -2\eta \frac{\partial J}{\partial g_1}, \Delta g_2 = -2\eta \frac{\partial J}{\partial g_2} \qquad (6)$$

where η is the step size or learning rate>0.Using these equations the parameters are updated.

4. Design of CFI based Controller using model obtained in Section 3

Once the plant is identified by way of fuzzy integral model, it is required to design a controller that can control the parameters of the identified plant. The model of the system designed using the above technique is used for finding the parameters of the controller. The block diagram of the complete system is shown in Fig. 3

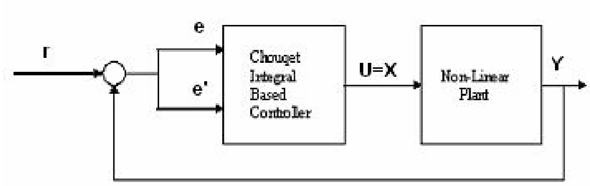

Fig 3 Block Diagram of the complete system

The learning algorithm used for Choquet Integral based controller is similar to that used for modeling. The controller takes error e (between the set point r and the actual output Y) and the change in error e' as input and updates the fuzzy densities. The densities are updated by using the same equations as in the modeling.

5. Implementation of CFI based Controller

After designing the CFI controller and simulating it on the identified model, the next step is to implement it on the plant. For this an interface was made between the hardware and the software. Since, the controller has been designed in the MATLAB, the interfacing with the hardware is also done in the same software. Though there are other softwares, which can be used for interfacing, but all those do not support controller design. On the other side MATLAB proves to be flexible as it supports the controller design and also supports many Data Acquisition (DAQ) cards. The main part of the interfacing is to send the output of controller to the control valve and to receive the actual value of the pressure so as to calculate the error between the desired and actual pressure. For sending and receiving the data to and from the PC, Data Acquisition (DAQ) cards have to be used. These DAQ cards are used for A/D and D/A conversion of the signals. All the Cards/Modules are manufactured by Advantech [7]-[8].

Three Cards/Modules used in the interfacing are:
i) ADAM – 4014D
ii) ADAM – 4520
iii) PCL – 726

(a) Description of cards/Modules

ADAM – 4014D

This card uses a 16-bit microprocessor controlled sigma – delta A/D converter to convert sensor voltage and current into digital data. It offers signal-conditioning, data display, A/D conversion, ranging, and high –low alarm and RS – 485 digital communication functions. It has two digital output and one digital input channel. This card is used to read the output pressure of the plant and send it back to the controller.

ADAM – 4520

It is used to convert RS-485 link to RS-232 link before sending the digital output value to the PC.

PCL – 726

This card provides six analog output channels on a single PC – BUS add on card. It is used to send the output voltage from the NN controller to the final control element (control valve).

(b) Position of the cards in the Plant

ADAM – 4014D is placed next to the pressure transmitter, which is recording the output pressure of the tank 2. The pressure sensor in the transmitter outputs a voltage signal (0 to 5 volts) to the ADAM – 4014D that converts this voltage value to the corresponding pressure value (0 to 100 psi). Now the digital value of the pressure is sent by ADAM – 4014D through RS – 485 communications.

The serial port on the PC can communicate only through RS -232 links. This means that the RS – 485 signal coming from the ADAM – 4014D has to be converted to RS – 232 format before sending it to the PC. Next ADAM – 4520 converts the RS – 485 signal coming from ADAM – 4014D to RS – 232 signal. So ADAM – 4520 is placed next to the ADAM – 4014D card [7].

PCL – 726 card is used to send the analog signals to E/P converter which converts this analog signal to the corresponding pressure signal. This pressure signal is then sent to the control valve. PCL – 726 card is placed just before the E/P converter. Fig. 3 and Fig. 4 describe the traveling of the signal using ADAM 4014D, ADAM 4520 and PCL 726.ADAM – 4014D converts the analog reading of the pressure transmitter into digital RS–485 values. This value is converted into digital RS-232 signal before transmitting it to COM 1 port of the PC. Hence the actual value of the plant is received, which is compared with the set point and the error becomes the input to the controller, which changes its parameters, so as to minimize it. The voltage output of the controller (0 to 5 Volts) is fed to the PCL – 726 card which is configured to give an output current having a range of 4-20mA (0V corresponds to 4mA and 5V corresponds to 20mA). From the card the current goes to the E/P converter, which converts 4–20mA into 3–15 psi pressure. This pressure is then applied to the pneumatic valve according to which the valve either opens or closes, and controls the pressure inside the tank 2, which is the actual pressure.

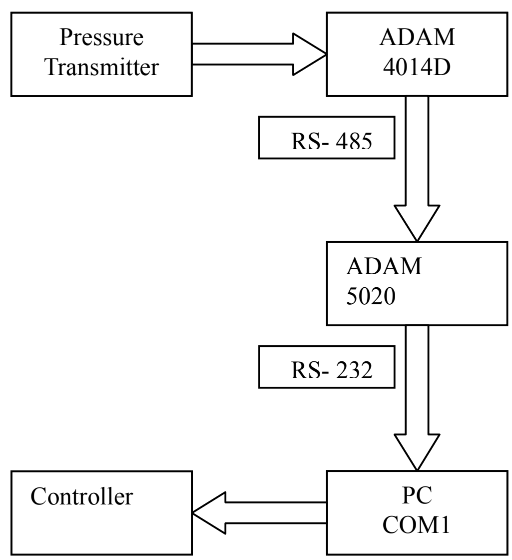

Fig. 3 Receiving the value of output pressure through serial communication

(c) MATLAB program to receive data from ADAM-4014D

A serial port interface program made in MATLAB helps to access various peripheral devices connected to the serial port of the computer. We have used

ADAM – 4014D module to read the value of the pressure in tank 2. ADAM – 4014D is connected to the serial port of the PC as it has to pass the value of the pressure to the controller (which is a program in MATLAB). To access this value of the pressure one should be able to read data from the serial port. Thus a serial port interface program is made in MATLAB.
The serial port interface program consists of following steps:-

(i) To construct the serial port object:
// S1 = serial ('COM 2', 'Baud Rate', 9600) ;
Above command creates a serial object S1. COM2 shows that the peripheral device is connected to COM port 2 of the computer Baud Rate of the serial object is set to 9600 kbps (i.e Baud Rate of the device connected).

(ii) To connect the serial port object to the serial port:
// fopen (S1)
Before using the serial port object to write or read data, it must be connected to the device via the serial port specified in the serial function. A serial port can be connected using the above command.

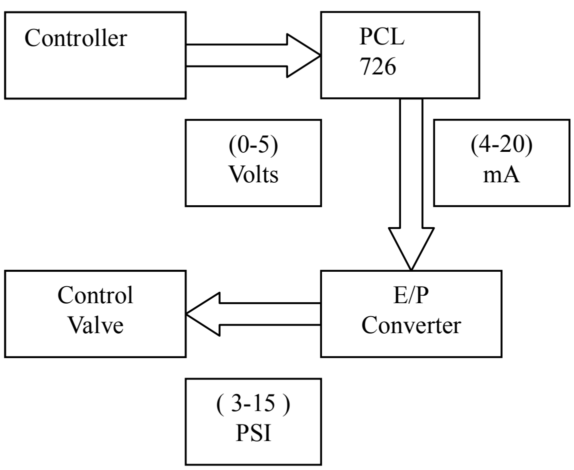

Fig. 4 Sending the output of controller to the control valve.

(iii) To receive data from ADAM – 4014D:
// fprintf (S1, '#AA');
To read the value of pressure from ADAM – 4014D value "#AA" is sent to ADAM via serial port with the help of above command. When ADAM receives "#AA" value, then it returns the digital value of the pressure to the computer.

(iv) To disconnect the serial object from the serial port:
// fclose (S1);
Once reading from the serial port is done, above command is used to end the serial port session.
Keeping in mind the above four steps a serial interface program was made in MATLAB to continuously read the value of the pressure in tank 2.

(d) Matlab program to send data from PCL -726:
We used Data Acquisition Toolbox ver2.5 in interfacing PCL-726 DAQ card to send voltage value to the control valve. MATLAB program follows the same steps as in the receiving.

6. Plant Response using CFI and FFNN Controllers

The first step for the control of the plant is to identify its model. Once the plant is identified, it is required to design a controller that can control the parameters of the identified plant. The designed controller is then simulated on the plant model and the response of the plant is observed. The performance of controller evaluated in terms of SE is also examined under parametric perturbations in which the densities of CFI based network used in the model are perturbed at the 10^{th} iteration. The controller is now implemented on the real time pressure feedback system, and its response is observed. Also a FFNN controller is simulated using the same model of the plant. Its parameters are also disturbed at 15^{th} iteration. Then the NN controller is implemented on the real time plant. The parameters of CFI based controller and FFNN controller were perturbed for comparison, when both attained the same SE though at different iterations

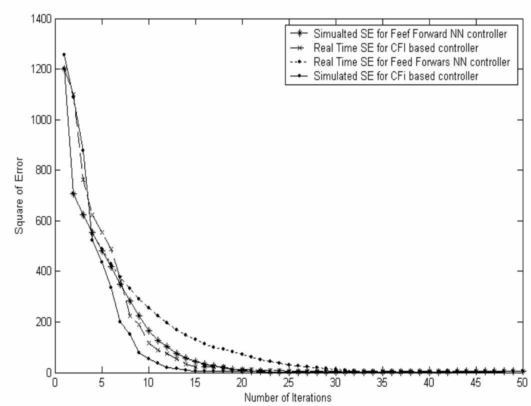

Fig. 5 SE using CFI and NN Controller

7. Results of Control and Discussion

The CFI based controller is successfully implemented on our real time system. The output pressure is digitally transmitted to the controller through ADAM-4014D. The voltage output of the controller in range 0-5V controlled the control valve opening through PCL-726, thereby controlling the pressure in the tank. The action of the CFI based controller is studied for several control schemes. Firstly, the controller is simulated on the plant model. Secondly, the densities of the plant model are perturbed and the CFI control action is observed. Next the controller is implemented on the real time system. Finally for comparison, Feed Forward Neural Network controller is simulated on the same plant model as well as implemented on the real time system. For better insight into the performance, its parameters are also perturbed.

Fig. 5 shows the results of control of real time plant as variation in SE under various schemes just mentioned. From Figure 5, the CFI based controller attains minimum SE in lesser number of iterations than FFNN controller both for simulation and implementation. Also the SE reduces faster and its minimum value is attained in lesser number of iterations when controller is simulated on the plant model than when it is implemented on the real time plant. This is valid for both the controllers.

Table 1 shows the minimum value of SE for CFI and FFNN controllers implemented on real time system along with minimum of SE for simulation. Both CFI and FFNN controllers can adapt to the nonlinearities of the real time system but CFI based controller adapts more readily because of non linear aggregation. Implementation of both controllers on real time system has certain limitations such as input-output constraints of each component, certain delay due to the inertia of the control valve, delay due to pressure transmitter etc. Therefore the implemented controllers have poorer minimum value of SE and attains at more number of iterations then simulated ones.

Table I

Mean Square Error	
CFI based Controller (Simulation)	1.09×10^{-3}
CFI base Controller (real time)	1.17×10^{-3}
FFNN(Simulation)	1.32×10^{-3}
FFNN(real time)	1.38×10^{-3}

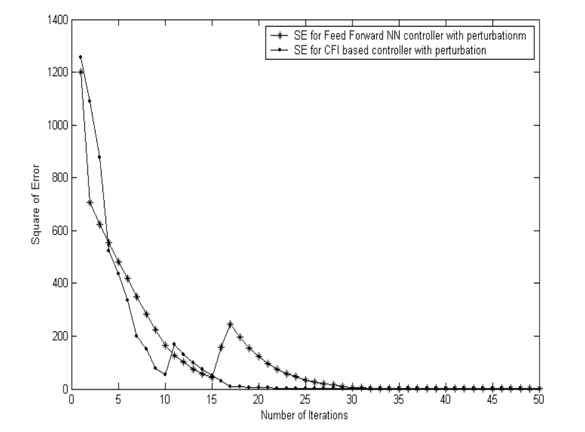

Fig. 6 SE using CFI and NN controller with perturbation

The parameters of the plant are perturbed for controllers when both have same value of SE. The effect of parameter perturbation is shown in Fig.6. For both the controllers, the SE increases when the parameters of the plant are perturbed. But this increase in case of FFNN is more than CFI based controller. Also from Fig. 6, the CFI based controller adapts faster to the disturbances than FFNN based Controller as the SE reduces readily in lesser number of iterations for CFI based controller than FFNN controller.

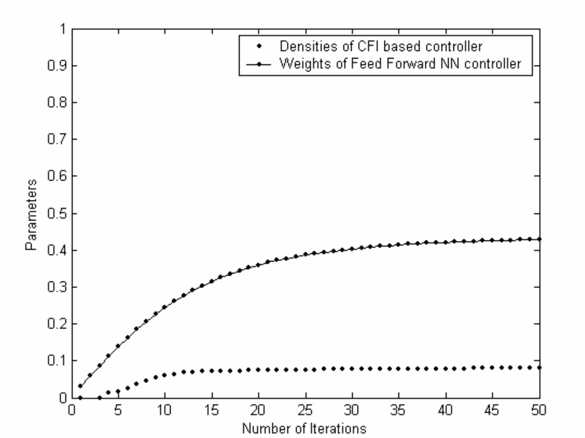

Fig. 7 Trained parameters for CFI and FFNN controllers

The parameters of the CFI based controller and FFNN controller are shown in Fig.7. The parameters of the controllers are trained through a learning rule. The parameters of the CFI based controller i.e. densities gets trained and causes the system to give the desired output in lesser number of iterations than the parameters of FFNN based controller i.e. weights.

8. Conclusion

The real time systems can generate outputs according to preset value using the CFI and FFNN controllers. Our pressure Feedback system adapted to the nonlinearities of the system readily for CFI based controller than FFNN controller. In comparison to FFNN controller, CFI based controller showed significant improvement with faster response towards the set point as it involves non linear aggregation. The CFI based controller is less influenced by parameter perturbation unlike FFNN controller. The learning efficiency of CFI based controller is superior to that of FFNN based controller. It is known that in case there are multiple solutions, CFI based controller picks up the optimal one. Because of these inherent characteristics, reality and preciseness of prediction with Choquet Fuzzy Integral based controller is more.

References

[1] E.H. Mamdani, "Application of Fuzzy algorithms for control of simpler dynamic plant", *Proc. of IEE*, vol. 121, no. 12, pp. 1585-1588, Dec 1974

[2] Chen-Wei Xu & Young-Zai Lu, "Fuzzy Model Identification and Self- Learning for Dynamic System", *IEEE Trans. On Systems Man and Cybernetics B,* vol. 17, no. 4, pp. 683-689, July/Aug. 1987.

[3] M. Sugeno, T. yasukawa, "A Fuzzy Logic based approach to qualititative modeling," IEEE Trans. On Fuzzy Systems, vol. 1, No. 1, pp 7-31, 1993

[4] J.M. Keller, P.D. Gader and A.K. Hocaoglu, " Fuzzy integrals in image processing and pattern recognition", chapter in fuzzy measures and Integrals, edited by M. Grabish, T. Murofusi and M. Sugeno, Berlin: Springerverlag,pp. 435-466,2000

[5] Jung-Hsien Chiang, "Choquet Fuzzy Integral-Based Hierarchical Networks for Decision Analysis," *IEEE Trans. On Fuzzy Systems*, Vol. 7, No. 1, pp. 63-71, February 1999.

[6] T.D. Phan and H.yan," Color image segmentation using fuzzy integral and mountain clustering," *Fuzzy Sets and Systems*, Vol. 107, pp.121-130,1999

[7 ADAM 4000 Series User's Manual, Ed. 5th: Advantech, 1997, pp. 4.1-4.99

[8] PCL-726 User's Manual. Available online: http://www.advantech.com