QUANTIZED CONTROLLER WITH CLOSE TO OPTIMUM NONLINEAR CONTROL SURFACE

Wing-Choi Ma and Bogdan M. Wilamowski
Department of Electrical Engineering
University of Wyoming
Laramie, WY 82071
email wing@uwyo.edu or wilam@uwyo.edu

ABSTRACT

The paper investigates a new kind of controller--the quantized controller, which can give a close to optimum nonlinear control surface. In contrary to the rough control surface generated by the fuzzy controller, the quantized controller can give better and smoother control. The quantized controller is trainable. By using an error function and the gradient decent method, control surface can be fine-tuned to some close to optimum control surface. The classical example of the backing truck to ramp problem is investigated using both the fuzzy controller and the quantized controller. Results show that the trained quantized controller can perform faster and smoother operations than the fuzzy controller.

I. INTRODUCTION

Fuzzy controllers are very successful in many practical applications . Unfortunately the control surface of a fuzzy controller is rough and step-like, this usually leads to coarse control. On the other hand, a neural network controller gives a smooth control surface, but the paths of the truck are sometimes irregular [1][2]. By using a mathematical function to smoothen the control surface and a training algorithm to fine-tune the surface to some close to optimum solution, the quantized controller can provide smooth and accurate control. The optimum control surface produced can be implemented by a PLA using the look-up table techniques [3]. As is shown in [3], this type of implementation has very fast response which has 8.3x10^6 FLIPS--Fuzzy Logic Inferences Per Second in comparison with 1x10^6 FLIPS using the ASIC approaches by Yamakawa [4]. Also, when the modified look-up table approach [3] for multiple inputs system is used to implement the quantized controller using microprocessor, it gives 15 times faster speed and also shorter codes than the fuzzy controller kernel designed by Motorola.

This paper investigates the quantized controller, the extrapolating technique, and the training algorithm. Finally, the quantized controller will be compared to the traditional fuzzy controller in the performance of backing up the truck. Results show that the quantized controller can park the truck smoother and quicker than the fuzzy controller.

II. EXEMPLARY OUTPUT PATTERNS

The domain of the input variables are separated into some segments. Similar to the fuzzy membership functions: the narrower the boundaries of segment, the finer is the control. In the example of backing up the truck presented in this paper, the x input variable is separated into 5 segments, and the ϕ input variable into 7 segments. As indicated in Figure 1, the segment closest to zero has the narrowest boundary to achieve better control. The required output values for these 48 input sets form 48 output patterns.

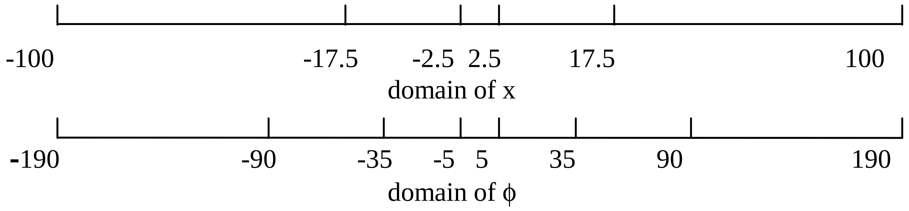

Figure 1. Boundary points of the input variables segments used to form the 48 input sets. (six boundary points for input x, and eight for the input ϕ).

The initial 48 known output patterns which contribute to the critical points on the quantized control surface can be obtained by the following ways:

1) get information from an expert or some experiments.
2) use results generated from a traditional fuzzy controller.

A good initial point can lessen the time required for the training process of the controller.

III. EXTRAPOLATING THE QUANTIZED CONTROL SURFACE FROM THE OUTPUT PATTERNS

A mathematical function is needed to extrapolate the control surface from the output patterns. As shown in Figure 2, Z1, Z2, Z3, and Z4 represent four critical points on the quantized control surface which form 4 nodes of a rectangular mesh. Points between nodes are obtained using the following equations:

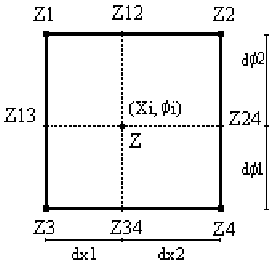

Figure 2. Variables for extrapolation

$$Z12 = \frac{Z1 \times dx2 + Z2 \times dx1}{(dx1 + dx2)} \qquad (1)$$

$$Z24 = \frac{Z2 \times d\phi 1 + Z4 \times d\phi 2}{(d\phi 1 + d\phi 2)} \qquad (2)$$

$$Z34 = \frac{Z3 \times dx2 + Z4 \times dx1}{(dx1 + dx2)} \qquad (3)$$

$$Z13 = \frac{Z1 \times d\phi 1 + Z3 \times d\phi 2}{(d\phi 1 + d\phi 2)} \qquad (4)$$

$$Z = \frac{Z12 \times d\phi 1 + Z24 \times dx1 + Z34 \times d\phi 2 + Z13 \times dx2}{(dx1 + dx2 + d\phi 1 + d\phi 2)} \qquad (5)$$

Other extrapolation techniques are also possible.

IV. THE TRAINING PROCESS

Traditional Fuzzy controllers usually require an expert who knows what the outputs should be for given input sets to design the FAM and the membership functions. With a trainable control system, the system can be trained to obtain optimum performance [5][6]. Training of fuzzy controller is possible but it do not give a smooth control surface. A training process is used by the quantized controller to fine-tune its initial control surface to a close to optimum control surface. The most important part of the training process is the proper definition of the error function. The error function acts as a supervisor for the training process. A good error function is very critical in finding the optimum control surface. In this backing up truck example, the following error function is used:

$$Er = \overset{Runs}{\underset{j=1}{}} (N_j + \Delta N_j) \qquad (6)$$

where Runs equals to 35 which is the number of test runs. A total of 35 different starting positions were used to test how well the controller control the truck. N_j is an integer value measuring the number of steps the truck needed to park to the ramp excluding the last step, and ΔN_j is a fuzzy value measuring what fraction of a step does the last move of the truck contains. ΔN_j is calculated using the following equation:

$$\Delta N_j = \frac{(Er_{end} - Er_{N_j})}{(Er_{N_{j+1}} - Er_{N_j})} \qquad (7)$$

where Er_{Nj+1} is defined by equation (8) and $Er_{Nj+1} <$ Er_{end} is the terminating criteria of a particular run.

$$Er_{N_{j+1}} = \left[x_{end}{}^2 + \left| \frac{|x_{\max}|}{|\phi_{\max}|} \times \phi_{end} \right|^2 \right| < Er_{end} = 1 \qquad (8)$$

Equations (6), (7) and (8) give a continuous analog value of Er.

The gradient descent method is used to train the quantized controller to a close to optimum control surface. The gradient descent method changes the whole quantized control surface at a time. The amount of change for the 48 output patterns is described by the following equation:

$$\Delta Z_j = -k \times (Er_j - Er_{init}) \tag{9}$$

where j ranges from 1 to 48 for the 48 different output patterns or the quantized control surface, and k is the learning constant. Er_j is the error obtained by equation (6) by increasing the output pattern Z_j by a small amount, and Er_{init} is the error of the initial quantized control surface. The difference between Er_j and Er_{init} gives the gradient information of the error function surface at point Z_j. All 48 output patterns are changed repeatedly by an amount described in equation (9) until the point where the local minimum of the error function surface is reached. This quantized control surface becomes the new initial quantized control surface, the new ΔZ_j is calculated, and the process continues.

V. COMPARISON BETWEEN FUZZY CONTROLLER AND THE QUANTIZED CONTROLLER

Simulations are done using the traditional fuzzy controller, the untrained quantized controller, and the trained quantized controller. The fuzzy controller uses the Kosko's data for the FAM and the membership functions [1][2]. The untrained quantized controller uses data generated by the same fuzzy controller to form the quantized control surface. The truck is started from 35 different initial positions -- the center points of the input membership functions of the fuzzy controller as described in [5] and both the path and the control surface for all three controllers are plotted for comparison as shown in Figure. 3 to 5. As the figures indicate, the trained quantized controller outperforms both the fuzzy controller and the untrained quantized controller in smoothness and accuracy in control. The trained quantized controller takes shorter time to stabilize to a small error than both the fuzzy controller and the untrained quantized controller. This is indicated by the large improvement in the values of the global error. The fuzzy controller takes the longest time out of the three to park the truck to the ramp. This is because of the extra winding paths of the truck which is a result of the stairs-like control surface. These results indicate that both the smoothing technique and the training algorithm proposed in this paper is successful in training the quantized controller for both high smoothness and accuracy in control.

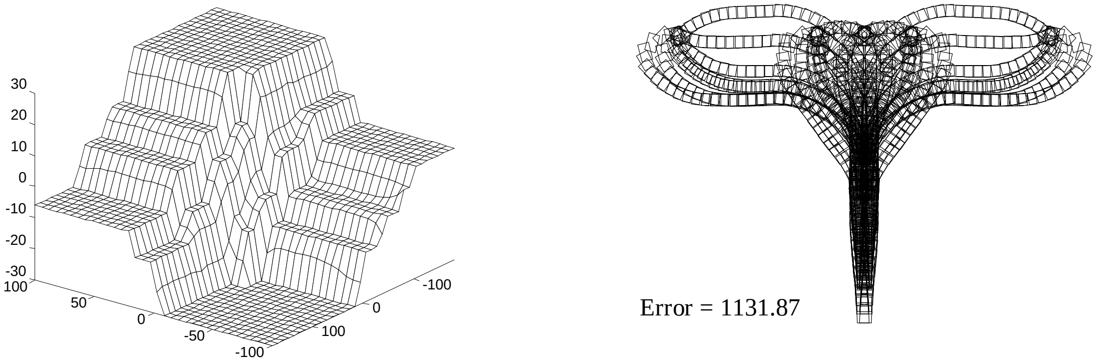

Figure 3. Control surface and traces of the truck obtained with a fuzzy controller

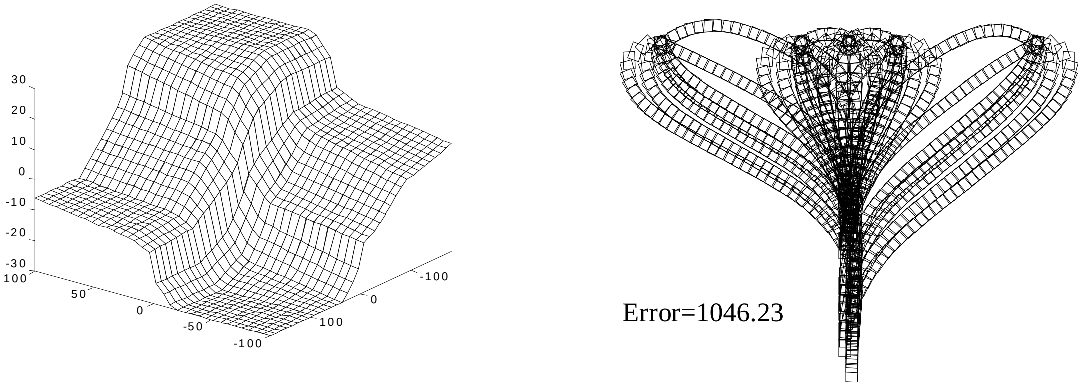

Figure 4. Control surface and traces of the truck obtained with an untrained quantized controller

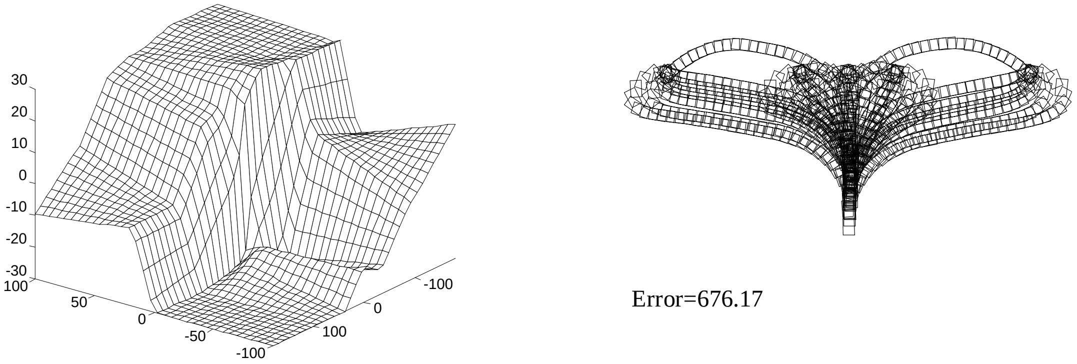

Figure 5. Control surface and traces of the truck obtained by a trained quantized controller

VI. CONCLUSION

The design and training of the quantized controller have been presented. Results show that the trained quantized controller can perform better than the traditional fuzzy controller. The extrapolating control surface and the training algorithm presented are successful in increasing the smoothness and accuracy in control. Both of these criteria leads to a close to optimum control surface which is better than the stairs-like rough control surface of the fuzzy controller. The quantized controller can then be implemented using the look-up table technique to achieve a fast response time or using a microprocessor based-approach which is simple and intuitive.

VII. REFERENCES

[1] Kosko, B., (1992). Neural Networks and Fuzzy Systems - A Dynamical Systems Approach to Machine Intelligence, Prentice Hall.

[2] Kong, S., Kosko, B., (1992). “Adaptive fuzzy system for backing up a truck-and-trailer,” IEEE Trans. on Neural Networks, (3), March, 211-223.

[3] Tan, H. S., Sandige, R.S., Wilamowski, B.M., (1994). “Hardware Implementation of PLD-based Fuzzy Logic Controllers Using a Look-up Table Technique,” Intelligent Engineering System Through Artificial Neural Networks, ASME Press, (4), 89-94.

[4] Yamakawa, T., (1988). “High-Speed Fuzzy Controller Hardware Systems: The Mega FLIPS Machine,” Information Science, Elsevier Science publishing Company, Inc., 113-128.

[5] Wilamowski, B. M., Sandige, R. S., (1993). “Trainable Fuzzy Controller,” Intelligent Engineering System Through Artificial Neural networks, ASME Press, (3), 561-566.

[6] Cupal, J.J., and Wilamowski, B.M., ,” Selection of Fuzzy Rules Using a Genetic Algorithm” proceedings of World Congress on Neural Networks, San Diego , California, vol1, pp. 814-819, June 4-9, 1994.