Fifth IFSA World Congress (1993), 1133-1136

# FUZZY CONTROLLER WITH MATRIX REPRESENTATION OPTIMIZED BY NEURAL NETWORKS

Mikio NAKATSUYAMA, Hiroaki KAMINAGA and Beidong SONG

Department of Electronic Engineering, Yamagata University, Yonezawa 992 Japan

**ABSTRACT:** Fuzzy algorithm is essentially nondeterministic, but to guarantee the stable control the fuzzy control program should be deterministic in practice. Fuzzy controllers with matrix representation is very simple in construction and very fast in computation. The value of the matrix is not adequate at the first place, but can be modified by using the neural networks. We apply the simple heuristic techniques to modify the matrix successfully.

## INTRODUCTION

Many researchers have been developing fuzzy controllers. There are many types of controllers. Some of them have a large number of parallel processors and execute the computation for control faster than even a super computer. Zadeh (1968) proposed fuzzy algorithm and showed an example of fuzzy automatic control of an automobile of which action is like a human performance. Yamakawa (1989) made the first fuzzy controller that controlled the inverted pendulum successively. Pappis (1977) used the fuzzy control statements for the traffic signal control and proved that his method is superior to other methods.
In general, fuzzy control is supposed to be nondeterministic. Fuzzy control program is constituted of many fuzzy control statements which are essentially the calculation of fuzzy minimum or and operator. The final decision is made by choosing the maximum values of these statements by using the simple maximum value. Though fuzzy program is based on the fuzzy sets or ambiguous concepts, fuzzy control program must provide only one determined value when some fixed inputs are applied to a fuzzy control system. If plural results are to be selected, then exact control can not be expected. Nakatsuyama (1990a) showed that the fuzzy program can be converted into matrices. At first, the value of the matrix is not adequate, so the fine adjustment of the value is required. We trained the neural networks enough to get the adequate value of the matrix representation. After combining the matrix representation, the neural network and the heuristic algorithm, we get the good result for controlling the traffics(Nakatsuyama 1992). The fuzzy controller with matrix representation optimized by the neural networks is simpler and is faster than the fuzzy control statements. In this paper, we improved this method by modifying the heuristic research. In application to the traffic control, the prediction of the queue improves the result especially when the traffic is very low.

## FUZZY CONTROL ALGORITHM

Fuzzy control statements are assembly of statements such as "if ... then ... else ... ". We show an example of fuzzy program which controls the traffic signals as follows. The terms mt and lt denote "more than" and "less than" respectively. In fact, the real calculation is executed as follows.

$$\mu = \max(\mu_R(t,a,q,e), \mu_S(t,a,q,e)...) \qquad (1)$$

The symbol $\mu$ denotes the membership function. The terms t, a, q and e are the normalized values of time, arrived vehicles, queue of vehicles and the signal duration respectively. If $\mu_R = \mu_S = ....$, we are able to choice any number as the most suitable value. The fuzzy control is nondeterministic in this sense. However, it is necessary to select only one value to guaranty the stable control. Then fuzzy control becomes deterministic. We get the following equation,

$$GP : f(t,a,q,e) \rightarrow e' \qquad (2)$$

The term e' is also the normalized value of the signal duration. In general, Eq.(1) can be written as follows.

$$GM : gm(k1,k2,k3,.....kn) \rightarrow em \qquad (3)$$

The term ki is an integer. Let M be a matrix, then this equation will become

$$e' = M(k1,k2,k3,...kn). \qquad (4)$$

The fuzzy control statements are represented as a matrix which we call the matrix representation. Of course, the gravity method (Mizumoto, 1991) always guaranty the unique output, but it does not mean that the fuzzy control algorithm is essentially deterministic.

## MATRIX REPRESENTATION OPTIMIZED BY NEURAL NETWORK

At first, the value of the elements of the matrix representation are supposed not to be optimal, so the fine adjustment of the values should be done before using the matrix for control. Nakatsuyama (1990b) proposed the method for the matrix modification which modifies the values of the matrix by the repetition of the control and by using the simple self-tuning. We got the good result, but we found that the correction is not sufficient. After several hundred training, we get the adequate values of weights. The output of the output layer is almost equivalent of the value of the matrix representation. The over all delay of the present and the past, the mean traffic, the mean value of the signal duration and the mean value of the queue are the input of the input layer . An input data set just corresponds one element of the matrix, so we determine the values of $5 \times 5$ or $3 \times 3$ elements at each input data set. To get more precise value, we use the mask matrix shown in Fig. 1.

| | | | | |
|---|---|---|---|---|
| 0.6 | 0.7 | 0.8 | 0.7 | 0.6 |
| 0.7 | 0.8 | 0.9 | 0.8 | 0.7 |
| 0.8 | 0.9 | 1.0 | 0.9 | 0.8 |
| 0.7 | 0.8 | 0.9 | 0.8 | 0.7 |
| 0.6 | 0.7 | 0.8 | 0.7 | 0.6 |

(a)

| | | |
|---|---|---|
| 0.8 | 0.9 | 0.8 |
| 0.9 | 1.0 | 0.9 |
| 0.8 | 0.9 | 0.8 |

(b)

Fig. 1. The mask matrix.

If a data set corresponds only one neuron, it does not need the mask matrix, while a large number of data sets are necessary. We use the function

$$out = 1 / ( 1 + \exp ( - p)), \qquad (5)$$

which is called as a simple sigmoidal activation function in the reference (Wasserman, 1989). The term p is the value of the output layer. After 500 training, we get almost the same value as the matrix representation by using 49 data sets which include the traffics 32, 64, 128, 256, 512,1024 and 1280 and cover all the necessary area. The term tr and tm are the traffics and the time duration of traffic signal respectively. The value of the matrix represents the queue q. The calculated time delay of the vehicles is 56.08. Both the trained data and the original data are normalized by 20.0. The value of the central element differs from the original data slightly, but does not have a bad influence upon the performance of the control.

## MODIFIED HEURISTIC ALGORITHM

The trained data simulate only the original matrix. Therefore it does not improve the performance of the control. Since the neural networks itself has not the ability of the reasoning, it is necessary to get this ability by using the training algorithm. Generally speaking, it is supposed that there is no explicit goal in fuzzy control. It is required only to get the better result than before.
We adopted the so-called heuristic algorithm to get the adequate value of the neural networks. The outline of the heuristic algorithm is as follows.

STEP 1

Determine the matrix and the temporary goal.
swt = 0. cv = 0.

STEP 2

swt = 1.

After simulating, the result i is obtained. Thereby the prediction is executed, if necessary.

STEP 3

If cv is 0, then swt = 1 and go to STEP 5.

STEP 4

If swt is 3 and the result i is better than result j, then the temporary goal becomes result i and the weights i of the neural networks is selected as better one and swt = 1 and cv = cv +1.

STEP 5

Calculate the output of the neural networks and the weight adjustment with the back propagation by using the temporary goal and result i.

STEP 6

If the difference between the last result and the next result is less than the predetermined small value $\epsilon$ , then stop.

STEP 7

If swt is 1, then the weights 1 of the neural networks are to be subtracted by the weight adjustment. If swt is 2, then the weights 2 of the neural networks are to be added by the weight adjustment.

STEP 8

Determine the matrix representation by using the modified neural networks. swt = swt + 1.
Go to STEP 2.

## FUZZY CONTROLLER

The matrix representation requires a lot of memories. Nowadays the price of the memory is not expensive, so we may use a large number of memory. Nakatsuyama (1990b) proposed the fuzzy controller with the matrix representation which is fast in computation. Our new proposition is based on the matrix representation modified by the neural networks which are improved by the simulation or experimental data(Nakatsuyama 1992). Its architecture is shown in Fig. 2. In this paper, we optimized the neural networks and use the more precise heuristic algorithm. Then we get the better results. The values of the matrix representation will be calculated by the repetition of the experiment or simulation. And the heuristic algorithm is very simple.

## APPLICATION TO TRAFFICS

We adopt the fuzzy controller to the traffic control. We suppose there are 8 one-way roads and there are 16 traffic junctions. Each traffic junction may be controlled by only one simple fuzzy controller which provide a good result. In fact, these 16 traffic signals are controlled by a single 20 $\times$ 20 matrix as shown in Fig. 3. If some one-way roads are designated to the artery road, the phase control is very effective. We show that a personal computer can control 16 traffic junctions easily. It is not effective to improve the time delay if the traffic is less than the 1024 vehicles / hour, since the values have already been precisely adjusted. So we can apply successively this method to the traffics control of which traffics is more than 1280 vehicles / hour. At first, we calculate the over-all delays of the vehicles when the prediction method is not executed. All the traffic junctions are controlled by a simple fuzzy controller with the matrix representation. It requires about 500 trainings to get the adequate weights of the neural networks. The output p of the neural networks is calculated by Eq. 6.

$$p(i,j) = \sum_k wk(i,j,k) \times ( sec(k,0) \times v + sec(k,1) \times w + sec(k,2) \times x + sec(k,3) \times y + sec(k,4) \times z ) \quad (6)$$

The terms wk and sec are both the weights of the neural networks. The term v, w, x, y, and z are the present delay time, the past delay time, the traffics, the queue, the signal duration respectively. The weight adjustment del is calculated by the following equations.

$$del(i,j) = \sum_k h \times ( temp - mean) \times ( sec(k,0) \times v + sec(k,1) \times w + sec(k,2) \times x + sec(k,3) \times y + sec(k,4) \times z ) \quad (7)$$

The values of sec(k,i) are are also adjusted by using del(i,j); By using the matrix representation modified by the refined neural networks and the simple heuristic search, we get the delay time 33.99 sec / vehicle at the traffics tr1 = tr2 = 1280, while the original delay time is 56.08. Then the delay time is improved by about 39%. Moreover we get the delay time 30.32, when we use the prediction of the queue. Then the delay time is still improved by about 11%. When we adopt the prediction of the queue, the delay time is improved by 20% even at the traffics tr1 = tr2 = 32.
If the waiting vehicle is succeeded directly by the next one and the line of the vehicles is not discontinued, the value of the queue q is added by 1. Naturally, the calculating of the queue stops when a waiting vehicle does not succeeded by another one. On the other hand, the prediction of the queue is calculated as follows. When the prediction is executed, the queue q is calculated by the following equation.

$$q = q + 1 \quad \text{if a succeeding vehicle exists.}$$
$$q = q + 1 / ( 1 + \log( 39 - i) ) \quad i \neq 39 \quad \text{if a succeeding vehicle does not exist.} \quad (8)$$

The term i means the number of the vehicles from the traffic junction though the vehicle may exists or not in fact. The number 39 means the maximum number of the waiting vehicles. The over-all delays are listed in Table 1. The delay time at 1536 vehicles / hour can not be simulated when the prediction is not executed.

## CONCLUSION

The matrix representation of the fuzzy controller is optimized by using the neuron networks, though it is necessary to use the simple and precise heuristic algorithm. The prediction of the queue of the vehicles is effective especially when the traffics are low. We showed the example of the control of the traffic signals and we got the good result. This method may be easily applied to any other control system. We acknowledge Mr. Mizunuma and Ms Sugimoto for their help.

## REFERENCES


Mizumoto, M. (1991). Min-max-gravity method versus product-sum-gravity method for fuzzy controls, *IFSA'91 Brussels*, (Engineering) pp.127-130.

Nakatsuyama, M., H.Nagahashi, N. Nishizuka and K. Watanabe (1990a). Matrix representation for fuzzy program and its application to traffic control, *11th IFAC WORLD Congress in Tallinn*, 7, pp.83-88.

Nakatsuyama, M., H. Nagahashi and K. Watanabe (1990b). Matrix representation for fuzzy programming and reconstitution of fuzzy program from modified matrix, *Proc. of Sino-Japan Joint Meeting on Fuzzy Sets and Systems*, Beijing.

Nakatsuyama, M.,H.Kaminaga (1992) Fuzzy controller with matrix representation modified by neural networks, *SICICA'92*, Malaga

Pappis, C.P. and E.H. Mamdani (1977). A fuzzy logic controller for a traffic junction, *IEEE Trans.Syst. Man Cybern.*, SMC-7,10,pp.701- 717

Wasserman, P.D. (1989). *Neural Computing Theory and Practice*, VAN NOSTRAND REINOLD, New York.

Yamakawa, T. (1989). Stabilization of an inverted pendulum by a high-speed fuzzy logic controller hardware system, *Fuzzy Sets and Systems*, 32, 2, pp.161-180.

Zadeh,L.A. (1968). Fuzzy algorithm, *Information and Control*, 12,pp.91-102.


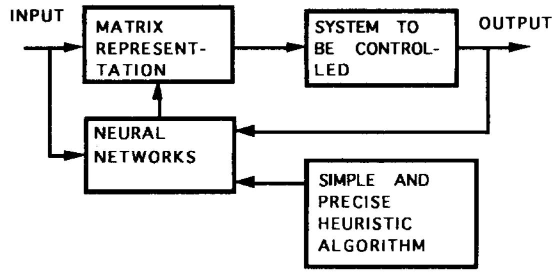


Fig. 2 Fuzzy controller with matrix representation optimized by neural networks.

tm

| | | 0 | 1 | 2 | 3 | 4 | . | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | 1 | 1 | 1 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 1 | 1 | 1 | 1 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 2 | 2 | 1 | 1 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 3 | 2 | 1 | 1 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 4 | 2 | 1 | 1 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 5 | 3 | 1 | 1 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 6 | 3 | 1 | 1 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 7 | 4 | 2 | 1 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 8 | 6 | 3 | 2 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 9 | 8 | 4 | 2 | 1 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| tr | 10 | 10 | 5 | 3 | 2 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 11 | 12 | 6 | 4 | 2 | 1 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 12 | 14 | 7 | 5 | 3 | 2 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 13 | 16 | 11 | 9 | 7 | 5 | . | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 14 | 17 | 15 | 13 | 10 | 8 | . | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 15 | 18 | 16 | 15 | 14 | 12 | . | 6 | 5 | 4 | 3 | 2 | 1 | 1 | 1 | 1 |
| | 16 | 19 | 18 | 17 | 16 | 16 | . | 8 | 7 | 6 | 5 | 3 | 4 | 2 | 1 | 1 |
| | 17 | 19 | 19 | 19 | 19 | 18 | . | 14 | 12 | 12 | 11 | 10 | 8 | 6 | 4 | 3 |
| | 18 | 19 | 19 | 19 | 19 | 19 | . | 16 | 14 | 14 | 13 | 12 | 10 | 10 | 8 | 6 |
| | 19 | 19 | 19 | 19 | 19 | 19 | . | 19 | 19 | 19 | 18 | 17 | 16 | 15 | 14 | 12 |

Fig. 3 The matrix representation.

Table 1 Over-all delays

| traffics | prediction | original |
|---|---|---|
| 32 | 2.34 | 3.14 |
| 64 | 2.78 | 3.64 |
| 128 | 3.34 | 4.15 |
| 256 | 4.54 | 5.25 |
| 512 | 6.02 | 6.96 |
| 1024 | 10.22 | 11.48 |
| 1280 | 30.32 | 56.08 |
| 1536 | 36.38 | ------- |