Fifth IFSA World Congress (1993), 1121-1124

Two-Degree-of-Freedom Fuzzy Neural Network Control System And Its Application To Vehicle Control

Satoshi Sekine, Toru Yamaguchi, Kouichirou Tamagawa and Tunekazu Endo

Systems & Software Engineering Laboratory, Toshiba Corp.
70, Yanagi-cho, Saiwai-ku, Kawasaki-shi, 210 JAPAN
Phone: +81-44-548-5637, Fax: +81-44-533-3593.
E-mail:sekine@ssel.toshiba.co.jp

ABSTRACT

We propose two-degree-of-freedom fuzzy neural network control systems. It has a hierarchical structure of two sets of control knowledge, thus it is easy to extract and refine fuzzy rules before and after the operation has started, and the number of fuzzy rules is reduced. In addition an example application of automatic vehicle operation is reported and its usefulness is shown by simulation.

1. INTRODUCTION

According to expert knowledge, to achieve intelligent control for varying dynamic characteristics and many purposes of control by means of synthesizing basic operation, we need not only *the most suitable steady-state control knowledge defined by the dynamic characteristics of the plant*, but also *control knowledge for changing the reference patterns defined by the control purpose*. But to express that control knowledge by means of a round robin combination, we have to write down a huge number of fuzzy rules [1],[2] that express complex conditions in the if-part and corresponding operations in the then-part. Moreover, appropriate operation knowledge corresponding to each condition is not always accurate and it is difficult to extract the control knowledge from experts before starting the operation.[3],[4]

Also, due to interference from fuzzy rules that have similar conditions, it can be difficult to refine the control knowledge after the operation has started.

To achieve the control purposes of a plant with dynamic characteristics which vary widely, we propose two-degree-of-freedom fuzzy neural network control systems. It has a hierarchical structure of two sets of control knowledge, thus it is easy to extract and refine fuzzy rules before and after the operation has started, and the number of fuzzy rules is reduced.

In addition an example application of automatic vehicle operation is reported and its usefulness is shown by simulation.

2. TWO-DEGREE-OF-FREEDOM FUZZY NEURAL NETWORK CONTROL SYSTEM

Fig.1 (page 4) illustrates the construction of a two-degree-of-freedom fuzzy neural network control system.

Besides the plant, there are two main parts to this system. One is a reference generator part that generates adequate reference patterns according to control purposes of the plant, the other is a stable controller part that puts out optimal actuating values according to the dynamic characteristics of the plant.

The reference generator part is consists of n pieces of reference generators (functions), a reference synthesizer, a purpose estimator, a fuzzy associative memory system and a reference generating rule trainer. Each reference generator i (i=1, 2, 3...n) puts out the typical reference pattern corresponding to its control purpose and the reference synthesizer interpolates them to make a general reference pattern which is suitable for the present control purpose. The purpose estimator calculates adequate performance indices to estimate the present control purpose. From these performance indices, the fuzzy associative memory system infers the parameter sets for each reference generator and the weight values of the reference synthesizer. The reference generating rule trainer refines the reference function selecting rules and the reference parameter tuning rules.

The fuzzy associative memory system has if-part membership functions, associative memories and a weight value and parameter synthesizer. Fuzzy inference is realized by means of memorizing the reference generating rules in the associative memory.

The stable controller part is basically the same structure as the reference generator part.

The two degree of freedom fuzzy neural network control system has several merits: The two sets of hierarchical control rules are independent and thus combination rules are not necessary. The fuzzy rules can be extracted from just the typical control purpose and the steady state conditions. Also, after the operation has started, it is easy to refine the rules. These merits make it easy to design a control system of this sort.

We now present equations in order to describe the system in more detail.

The reference generators are linear or non-linear functions that have several parameters. These are given by

$$\mathbf{r}_i^* = \mathbf{f}_{ri}(\mathbf{P}_{ri}, \mathbf{y}) \qquad (i=1,2,3\cdots n). \tag{1}$$

Where $\mathbf{r}_i^*$ is the reference value according to each control purpose, $\mathbf{f}_{ri}$ is a linear or non-linear function, $\mathbf{P}_{ri}$ represents the parameter sets, y represents the state variables of the plant. (Boldface type denotes vector quantities.)

The reference synthesizer multiplies each reference value by a corresponding weight value W_{ri} and adds all the reference values up, yielding $\mathbf{r}^*$

$$\mathbf{r}^* = \sum_{i=1}^{n} W_{ri}\mathbf{r}_i^*. \tag{2}$$

The purpose estimator calculates the performance indices (J_{rf}, J_{rp}) so the reference generators (functions) can be selected and the reference function parameters can be fixed.

The fuzzy associative memory calculates the certainty value of each fuzzy label from the if-part membership functions, then draws an inference to decide the activation of the associative memory a_{rfi} according to a set of reference function selecting rules such as

$$\text{Rule 1: if } J_{rf} \text{ is } L_{rf1} \text{ then } r^*=f_{r1}(P_r, y) \text{ with } C_{rf1},$$
$$\cdots$$
$$\text{Rule n: if } J_{rf} \text{ is } L_{rfn} \text{ then } r^*=f_{rn}(P_r, y) \text{ with } C_{rfn}. \quad (3)$$

Where L_{rfi} is a fuzzy label and C_{rfi} uses a value $(\in[0,1])$ representing the connection rate between the reference function $f_{ri}()$ and the optimum reference function $f_r^*()$.

The fuzzy associative memory also draws an inference to decide the activation a_{rpi} for each function's parameter according to the set of reference parameter tuning rules.

Finally the weight and parameter synthesizer yields the weight value W_{ri} and parameter sets P_r as follows.

$$W_{ri} = a_{rfi} / \sum_{i=1}^{n} a_{rfi}. \quad (4)$$

$$P_r = \left\{\sum_{i=1}^{n} a_{rpi} K_{ri}\right\} / \left\{\sum_{i=1}^{n} a_{rpi}\right\}. \quad (5)$$

The stable controller part produces an activation vector u^* and infers the parameter sets P_c for the stable controllers and the weight value W_{ci} for each activation value synthesizer, all in the same manner as the reference generator part.

3. APPLICATION TO THE AUTOMATIC VEHICLE OPERATION

A transit system [5],[6] is subject to widely varying external conditions such as weather, the time of day, etc. Different situations imply different control purposes and the varying conditions induce fluctuating dynamic characteristics. Therefore it is difficult to automatically control a vehicle satisfactorily.

In this paper we apply a two-degree-of-freedom fuzzy neural network control system to automatic vehicle operation. The controller's function is to accurately stop a vehicle at a station. We investigate the controller's performance with acceleration error due to changing dynamic characteristics and with a control purpose transition from velocity control to position control.

Suppose that the vehicle is moving straight at acceleration α_r. The motion equations for velocity v, position x and time t of the vehicle are given by

$$v=\alpha_r(t-t_0)+v_0, \quad (6)$$
$$x=\frac{1}{2}\alpha_r(t-t_0)^2+v_0(t-t_0)+x_0. \quad (7)$$

Where x_0, v_0 are position and velocity at time t_0.

From equations (6) and (7), we eliminate the time t and obtain the equations (8) and (9).

$$v=\sqrt{2\alpha_r(x-x_0)+v_0^2}, \quad (8)$$
$$x = (v^2-v_0^2)/2\alpha_r + x_0. \quad (9)$$

Then we use eq.(8), (9) as reference functions for velocity and position control respectively.

To estimate the transition of these two controls, we use T_r, the time remaining until the vehicle has stopped at the station as the performance index. Equation (10) is derived from (6) and (7) by replacing x_0 with x_s, setting v_0 to 0, and replacing $t-t_0$ by T_r.

$$T_r = 2(x_s-x)/v. \quad (10)$$

T_r is used to switch the control purpose between velocity control and position control. This is decided by the following fuzzy rules:

$$\text{Rule 1: if } T_r = TB \text{ then } \begin{Bmatrix} \alpha_v^* = \alpha_r \\ v_v^* = \sqrt{2\alpha_r(x-x_0)+v_0^2} \\ x_v^* = x \end{Bmatrix} \text{ with } C_{r1},$$

$$\text{Rule 2: if } T_r = TS \text{ then } \begin{Bmatrix} \alpha_x^* = \alpha_r \\ v_x^* = v \\ x_x^* = (v^2-v_0^2)/2\alpha_r + x_0 \end{Bmatrix} \text{ with } C_{r2}. \quad (11)$$

Where TB and TS are fuzzy labels, α_v^*, v_v^*, and x_v^* are acceleration, velocity and position reference for velocity control, α_x^*, v_x^*, x_x^* are acceleration, velocity and position reference for position control. This also decides the activation a_{ri}.

The weight values for velocity control W_v and position control W_x are given by

$$W_v = a_{r1} / (a_{r1}+a_{r2}), \quad (12)$$
$$W_x = a_{r2} / (a_{r1}+a_{r2}). \quad (13)$$

The reference synthesizer multiplies the each reference value by the weight values W_v and W_x and sums them up to make general reference values α^*, v^* and x^*

$$\alpha^*=W_v\alpha_v^*+W_x\alpha_x^*, \quad (14)$$
$$v^*=W_v v_v^*+W_x v_x^*, \quad (15)$$
$$x^*=W_v x_v^*+W_x x_x^*. \quad (16)$$

In the stable controller part, we subtract the state variables v and x from the reference values v^* and x^* respectively and compute the errors $\Delta v, \Delta x$

$$\Delta v=v^*-v, \quad (17)$$
$$\Delta x=x^*-x. \quad (18)$$

The characteristic estimator, an acceleration error detector, calculates the acceleration error $\Delta\alpha$ from the velocity reference v^* and the detected velocity v.

Each stable controller i (i=1,2,3), adapted to each typical steady-state, consists of PI-controllers that have proportional gains G_{vpi} for velocity control and G_{xpi} for position control, integral gains G_{vIi}, G_{xIi}, and also have feed-forward gains G_{vfi}, G_{xfi} for the reference acceleration α^*.

The fuzzy associative memory infers to decide the activation a_{ci} of each controller gain according to the following rules.

$$\text{Rule 1: if } \Delta\alpha \text{ is AP then } \left\{ \begin{array}{l} G_{vf1}=K_{vf1}\ G_{xf1}=K_{xf1} \\ G_{vP1}=K_{vP1}\,G_{xP1}=K_{xP1} \\ G_{vI1}=K_{vI1}\ G_{xI1}=K_{xI1} \end{array} \right\} \text{ with } C_{cp1},$$

$$\text{Rule 2: if } \Delta\alpha \text{ is AZ then } \left\{ \begin{array}{l} G_{vf2}=K_{vf2}\ G_{xf2}=K_{xf2} \\ G_{vP2}=K_{vP2}\,G_{xP2}=K_{xP2} \\ G_{vI2}=K_{vI2}\ G_{xI2}=K_{xI2} \end{array} \right\} \text{ with } C_{cp2},$$

$$\text{Rule 3: if } \Delta\alpha \text{ is AN then } \left\{ \begin{array}{l} G_{vf3}=K_{vf3}\ G_{xf3}=K_{xf3} \\ G_{vP3}=K_{vP3}\,G_{xP3}=K_{xP3} \\ G_{vI3}=K_{vI3}\ G_{xI3}=K_{xI3} \end{array} \right\} \text{ with } C_{cp3}. \quad (19)$$

where AX (X=P, Z, N) is the fuzzy label and $K_{vfi}, K_{xfi} \ldots K_{xIi}$ (i=1,2,3) are the constants of each gain. C_{cpi} represents the connection rate between the constants of gains $K_{vfi} \ldots K_{xIi}$ and the optimal gains $K^*_{vfi} \ldots K^*_{xIi}$.

The weight synthesizer compose the weight value W_{ci} for each stable controller by eq.(20).

$$W_{ci} = a_{ci} / \sum_{i=1}^{3} a_{ci}. \quad (20)$$

Then we obtain the actuating value $\hat{\alpha}$ as follows.

$$\hat{\alpha} = \sum_{i=1}^{3} W_{ci} \Big(G_{vPi}\Delta v + G_{vIi} \int \Delta v dt + G_{vfi}\alpha^* + G_{xPi}\Delta x + G_{xIi} \int \Delta x dt + G_{xfi}\alpha^* \Big). \quad (21)$$

When we refine the rules, we add a corrective actuation value Δu_t to $\hat{\alpha}$ and make the actuating value α_u. Finally we convert α_u to a notched actuating value α_N and apply α_N to the vehicle.

We refine the the respective controllers by training the running operation. Before learning the operation gain, we use a gain which is not a good fit for each condition but which is robustly stable for all conditions.

By the learning algorithm, each uncertain operation gain is individually trained under the condition and the relationship between each condition and its operation gain is reinforced. We use the Widrow-Hoff (1960) learning algorithm to refine each operation gain under its condition.

The operation error E is given by

$$E = \sum_{t=1}^{T} \Big(\Delta u_t - \Delta G_{vf1}\alpha^*_t - \Delta G_{vP1}\Delta v^*_t - \Delta G_{vI1} \int \Delta v^*_t dt \Big)^2 \rightarrow \min. \quad (22)$$

In (22), Δu_t is the training signal of the operator, and $\Delta G_{vf1}, \Delta G_{vP1}$ and ΔG_{vI1} are the refined values of the stable controller 1's gain for the velocity control. We would like to minimize E. ΔG_{vP1} is given by

$$\Delta G_{vP1} = \sum_{t=1}^{T} \Delta u_t \Delta v_t \Big/ \sum_{t=1}^{T} \Delta v_t^2. \quad (23)$$

We refine each controller's gain in the same fashion as G_{vP1}.

4. SIMULATION RESULT

To verify the performance of the control system, we simulate it according to the following three conditions.

1. The vehicle is controlled only by velocity control without gain selecting.
2. The vehicle is controlled by both velocity and position control without gain selecting.
3. The vehicle is controlled by both velocity and position control with gain selecting (proposed system).

In each simulation we give the vehicle an acceleration error and investigate the stopping position error and the number of actuating value changes. The initial velocity of the vehicle is 55km/h. When the vehicle enters the deceleration section, a force is applied to produce a deceleration of α^*_r=-2.11 [km/h/sec], so that the vehicle can be controlled to stop at 350[m] from the start point. The running resistance, a parameter of the vehicle, is $T_L=2.3+4.8\times10^{-3}v+7.46\times10^{-4}v^2$ [kgf/ton] and the delay time before starting deceleration is T_d=2.0 [sec]. We produce the vehicle's acceleration error by changing the vehicle's mass. The gains for each stable controller are shown in table 1. These gain values were derived by trial and error.

Table 1. Stable controller's gain

	Controller 1 Acceleration error -30%	Controller 2 Acceleration error 0%	Controller 3 Acceleration error +30%
K_{vf}	1.40	1.00	0.80
K_{vP}	0.30	0.20	0.15
K_{vI}	0.03	0.02	0.015
K_{xf}	1.20	1.00	0.80
K_{xP}	2.00	1.60	2.00
K_{xI}	1.00	0.80	1.00

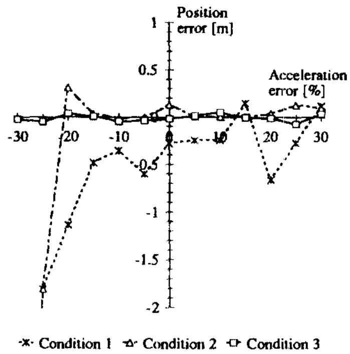

Fig.2 Stopping position error

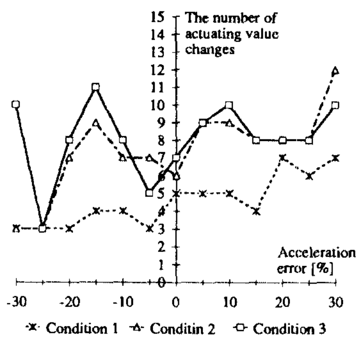

Fig.3 The number of actuating value changes

Fig.2 shows the stopping point error for acceleration errors from -30% to +30%.

Under condition 3, the vehicle can stop within an absolute stopping position error of 0.1 [m]. This is because after deceleration starts, the gain of the stable controller is adjusted to the most suitable value by fuzzy inference according to the acceleration error.

Fig.3 shows the number of actuating value changes versus acceleration error. The number under conditions 2 and 3 is higher compared to the number under condition 1. Even though the number of changes has increased, the high gain of the stable controller keeps the system stable.

Fig.4 shows the transition of velocity, velocity error, position, position error and activation value with an acceleration error of -30% under condition 3. The vehicle produces a weaker braking force as was expected, resulting in the velocity error becoming largely negative. The controller recognizes that the acceleration error is negative and changes the initial gain into a big gain and increases the actuating value. Near the stopping point, the more the position controller puts out the actuating value, the less the velocity controller does. The transition from velocity control to position control is smooth.

5. CONCLUSION

We proposed a two-degree-of-freedom fuzzy neural network control system and we illustrated that it was easy to extract and refine fuzzy rules before and after the operation has started, and that the number of fuzzy rules required is reduced by this technique.

In addition we reported an example application of this system to automatic vehicle operation and showed that the vehicle was controlled well in spite of changes in control purpose and dynamic characteristics.

In this paper, we only simulated the test cases without learning, but we will execute the simulation and make an experiment on those cases with learning soon.

REFERENCES

[1]E.H.Mamdani and S.Assilian:"An experiment in linguistic synthesis with a fuzzy logic controller",International Journal of Man-Machine Studies,7,1,pp.1-13(1975).

[2]T. Terano,"Fuzzy System Theory and Its applications", 1987, Ohbunsha (in Japanese).

[3]T. Yamaguchi et al.:"Intelligent Control of A Flying Vehicle Using Fuzzy Associative Memory System", *IEEE International Conference on Fuzzy Systems 1992 proc.pp.1139-1149* .

[4]S.Hayashi, "Auto-tuning Fuzzy PI Controller", *IFSA '91, conf. proc. E1, pp.41-44.*

[5]H.Takahashi et al. "Subjective Evaluation Modeling Using Fuzzy Logic And A Neural Network", *3rd IFSA '89 cong. proc. pp.520-523.*

[6]S.Yasunobu, et. al. "Fuzzy Control for Automated Train Operation System", *4th IFAC/IFIP/IFRS Int. Conf. on Transportation Systems, pp.39-45(1983).*

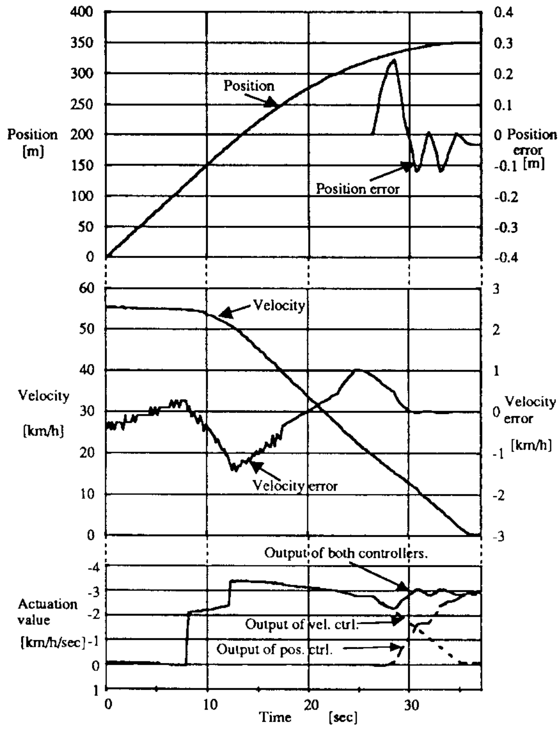

Fig. 4 Transition of velocity, velocity error, position, position error and actuating value

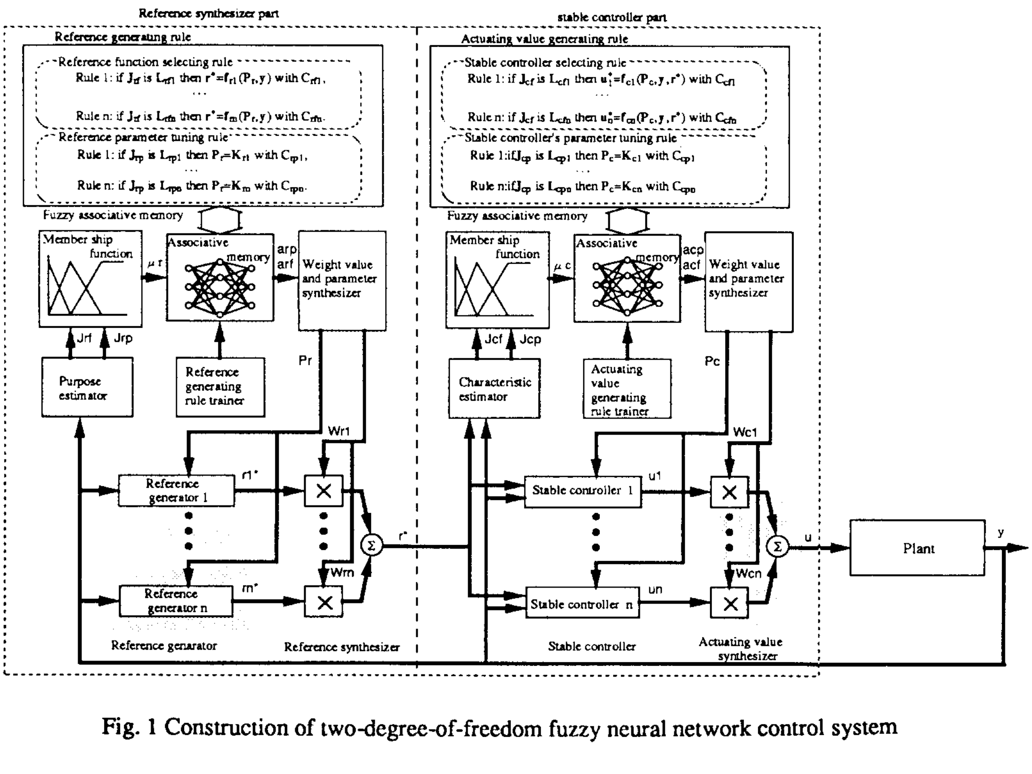

Fig. 1 Construction of two-degree-of-freedom fuzzy neural network control system