Multiple Fuzzy Controls*

Keigo WATANABE**, Kozo SHIRAMIZU
and Toshio FUKUDA***

A multiple fuzzy controller is proposed for self-organizing fuzzy control in which several "elemental fuzzy controllers" are processed in parallel and the degree of usage of each inferred consequent is determined by using a linear neural network. The inputs to the neural network are all the inferred consequents generated from the elemental fuzzy controllers, and the output of the neural network is the control input to the plant. The delta rule is used to update the connection weights for the network so that the square of plant output deviation is minimized. The present approach allows the elemental fuzzy controller to be used in situations in which the controller parameters are tuned incompletely; thus the number of trials and errors required for tuning the parameters can be decreased significantly. The effectiveness of the present fuzzy controller is illustrated by a simulation for the attitude control of a flexible satellite.

Key Words: Automatic Control, Computer Control, Identification, Iterative Learning Control, Fuzzy Control, Multiple Controllers, Neural Networks, Self-organizing

1. Introduction

Fuzzy control is attractive as a practical control strategy, because it does not require strict mathematical models and can easily apply it to nonlinear systems as well as linear systems, unlike the conventional optimal control. It should be noted, however, that, when designing a fuzzy controller, we must suitably (or optimally) tune some controller parameters such as scalars for the input data in the antecedent and the reasoning consequent, the number of membership functions, the width of a membership function, and the number of control rules, etc., through considerable trials and errors. For these problems, so-called self-organizing fuzzy controllers (SOFCs)

---

** Department of Mechanical Engineering, Faculty of Science and Engineering, Saga University, 1 Honjo-machi, Saga 840, Japan
*** Department of Mechano-Informatics and Systems, Faculty of Engineering, Nagoya University, Furocho, Chikusa-ku, Nagoya 464-01, Japan

learning law to the inverse dynamics model that was originally used by Kawato et al.\(^{(19)}\) and by replacing the original neural network component with Sugeno’s fuzzy reasoning\(^{(21)}\). However, this proposed method consists of multistep learnings, so it consequently has drawbacks of increasing memory capacity and computation time. Hayashi et al.\(^{(11)}\) presented a method for determining fuzzy sets in the antecedent and input-output relation in the conclusion, without finely tuning the reasoning rules, by using the learning ability of neural networks with actual input-output data of the plant. In this method, the input-output data of the plant for the learning (or testing) must be divided with respect to the reasoning rules. It also needs more repetitions of learning to identify the structure of the antecedent and conclusion. In addition, one has to gather the input-output data, as training data, obtained from a typical control pattern. Furthermore, in a study by Horikawa et al.\(^{(12)}\), the structure of a neural network itself is represented as a kind of fuzzy control rule, which is advantageous in that the control rules can be identified from the unknown states. Like the method of Hayashi et al.\(^{(11)}\), this method must also use the data, as training data, obtained from several manipulated patterns by an expert, so it is not necessary that it be applied widely.

In this paper we propose a multiple fuzzy controller in which fundamental concepts in the conventional fuzzy controller are followed, several fuzzy controllers with incompletely tuned parameters are processed in parallel, and the degree of the usage of inferred consequent from each “elemental fuzzy controller” is learned by a linear neural network. In this type of fuzzy controller, it is sufficient to tune the parameters of the elemental fuzzy controllers so that it gives a moderate or stable control result. Therefore, we can drastically decrease trials and errors that are required to obtain a better or approximately optimal control result by using a conventional single fuzzy controller. Moreover, the present fuzzy controller has much simpler structure than a learning-type fuzzy controller with the neural networks by Watanabe and Ichihashi\(^{(20)}\) or Hayashi et al.\(^{(11)}\) described above.

In the following, we first explain the elemental fuzzy controller using simplified fuzzy reasoning, in which the conclusion consists of a constant value. We then describe the construction method of the multiple fuzzy controller and the associated learning algorithm of a linear neural network that minimizes the squared norm of the output deviation of the plant. Finally, we consider the attitude control of a critically stable artificial satellite and illustrate the effectiveness of the proposed approach by some simulations.

2. Elemental Fuzzy Controller

The multiple fuzzy controller consists of several fuzzy controllers, in which one fuzzy controller is called an “elemental fuzzy controller.” This elemental fuzzy controller is described by “simplified fuzzy reasoning”\(^{(14)}\), whose reasoning is an approximation of Mamdani’s method\(^{(15,16)}\); i.e., the conclusion has a constant value. Following this method, we can write any \(i\)-th control rule as

\[
R_i: \text{If } x_1 = A_{i1} \text{ and } \ldots \text{ and } x_n = A_{in} \text{ then } u_1 = B_{i1} \text{ and } \ldots \text{ and } u_p = B_{ip}
\]

(1)

for \(n\)-input variables \((x_1, \ldots, x_n)\) and \(p\)-output variables \((u_1, \ldots, u_p)\). Here, \(R_i\) is the \(i\)-th control rule, \(A_{ij}\) is the fuzzy set (or fuzzy variable) of the antecedent part corresponding to the \(j\)-th input variable in the \(i\)-th control rule, and \(B_{ij}\) is the constant value corresponding to the \(j\)-th output variable in the \(i\)-th control rule. Using \(n\) confidences \(\mu_{A_{ij}}(x_j), \ldots, \mu_{A_{in}}(x_n)\), we have the confidence of the antecedent part in the \(i\)-th control rule, \(h_i\), as follows:

\[
h_i = \mu_{A_{i1}}(x_1) \land \mu_{A_{i2}}(x_2) \land \ldots \land \mu_{A_{in}}(x_n)
\]

(2)

where \(\land\) denotes the minimum operator. Then, the \(j\)-th inferred consequent is given by

\[
u_j = \frac{\sum_{i=1}^{r} h_i B_{ij}}{\sum_{i=1}^{r} h_i}, \quad j=1, \ldots, p
\]

(3)

where \(r\) denotes the total number of control rules; generally \(r = I^p\) if the number of labels is \(I^p\)

Figure 1 shows a fuzzy set consisting of 7 labels for two inputs \((x_1, x_2)\) and single output \((u^t)\), in which the width of the triangular membership function is 4 and the support set is \([-6, 6]\). Then, the corresponding control rule table is given, for example, as Table 1. Here, each label represents

<table>
<thead>
<tr>
<th>(x_1)</th>
<th>(x_2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(NB)</td>
<td>(-6)</td>
</tr>
<tr>
<td>(NM)</td>
<td>(-6)</td>
</tr>
<tr>
<td>(NS)</td>
<td>(-4)</td>
</tr>
<tr>
<td>(ZO)</td>
<td>(-6)</td>
</tr>
<tr>
<td>(PS)</td>
<td>(-4)</td>
</tr>
<tr>
<td>(PM)</td>
<td>(2)</td>
</tr>
<tr>
<td>(PB)</td>
<td>(4)</td>
</tr>
</tbody>
</table>

Fig. 1 Fuzzy set consisting of 7 labels
NB : negative big
NM : negative medium
NS : negative small
Z0 : zero
PS : positive small
PM : positive medium
PB : positive big

Note also that these rules are based on using the most commonly utilized fuzzy rules, in which the conclusion is a fuzzy set, but only after modifying the rules corresponding to the right-upper and left-lower parts in the table, and letting the constant value in the conclusion be a value on the support set where the original membership function has the confidence of unity.

When this type of fuzzy set is used, to transform the input variables to the values on the support set \([-6, 6]\), we need the scalers before and behind the fuzzy controller. Here, we define that GIN(i), \(i = 1, \ldots, n\), and GOU(i), \(i = 1, \ldots, p\), are the input scalers and the output scalers for the inferred consequents \(u^t, \ldots, u^p\). A block diagram of an elemental fuzzy controller is shown in Fig. 2. Although the defuzzification used here is obtained as a result of position-type reasoning, we can also use the result of velocity-type reasoning.

3. Multiple Fuzzy Controller

In order to construct the proposed multiple fuzzy controller, consider the following nonlinear plant with \(p\)-input and \(m\)-output vectors

\[ x(t) = f(x(t), u(t), t) \]  
\[ y(t) = g(x(t), t) \]

where \(x(t)\) is the \(s\)-dimensional vector, \(u(t)\) is the \(p\)-dimensional input vector, \(u(t) = [u_1(t), \ldots, u_p(t)]^T\), \(t\) is the continuous time variable, \(f(\cdot)\) is an \(s\)-dimensional nonlinear vector-valued function, \(g(\cdot)\) is an \(m\)-dimensional nonlinear vector-valued output function and \(y(t)\) is the \(m\)-dimensional plant output vector \(y(t) = [y_1(t), \ldots, y_m(t)]^T\).

3.1 Construction method

When defining the reference vector as \(y_d(t)\), we use the output deviation

\[ e(t) = y_d(t) - y(t) \]

and the corresponding \(e(t)\) (or increment \(\Delta e(kT)\)) as the inputs to the antecedent part of the fuzzy reasoning, where \(\Delta = 1 - z^{-1}\), \(z^{-1}\) is the one-step delay operator, \(k\) is a positive integer and \(T\) is a sampling width. We then drive \(M\) elemental fuzzy controllers in parallel with incompletely tuned controller parameters, e.g., the number of labels, \(i\), for the antecedent and conclusion, the width of membership function, the maximum and minimum values of the support set, \([-L, L]\), the number of control rules, \(r\), and the input and output scalers, GIN(i), \(i = 1, \ldots, n\) and GOU(i), \(i = 1, \ldots, p\). Consequently, we can obtain the inferred consequents \(u_i(t) = [u^i_1(t), \ldots, u^i_p(t)]^T\), \(i = 1, \ldots, M\) from each elemental fuzzy controller. By multiplying this inferred consequent by weight \(w_i\), the actual control input to the plant can then be generated by

\[ u(t) = \sum_{i=1}^{M} w_i u_i(t). \]  

Figure 3 shows the block diagram of a control system using the multiple fuzzy controller. Note that the neural network with \(\{u_i, u\}\) as the input and output relation does not use any sigmoidal function, so we call it a linear neural network.

3.2 Learning method

By letting the output from each elemental fuzzy controller be an input to the linear neural network and learning the connection weights \(w_i\) of the network, the present multiple fuzzy controller tries to obtain a much better control input than that obtained from a single fuzzy controller with any untuned controller parameters. Therefore, for the learning of \(w_i\), we introduce a cost function consisting of the squared norm of the output deviation,

\[ J = \frac{1}{2} | e(t) |^2 \]

and construct the learning algorithm based on minimizing this function. From Eqs. (6)-(8), the gradient of \(J\) with respect to \(w_i\) becomes

\[ \frac{\partial J}{\partial w_i} = -e^T \frac{\partial y}{\partial u} \frac{\partial u}{\partial w_i} \]

\[ = -e^T \frac{\partial y}{\partial u} u_i. \]

Therefore, using the delta rule \(\Delta w\) gives the following

Fig. 2 Block diagram of an elemental fuzzy controller

Fig. 3 Block diagram of a multiple fuzzy control system
update equation for $w_i$:

$$w_i(l+1) = w_i(l) + \eta e^T \frac{\partial y}{\partial u} u_i$$  \hspace{1cm} (10)$$

where $l$ denotes the $l$-th update time and $\eta$ is a small positive constant. The Jacobian $\frac{\partial y}{\partial u}$ can be correctly obtained, if the plants (4) and (5) are known. But, in the application of a fuzzy control the plant is assumed to be unknown, so we will approximately compute it as

$$\left[ \frac{\partial y_i(t)}{\partial u_i(t)} \right]_{i=1, \cdots, m, j=1, \cdots, p} \approx \left[ \frac{\Delta y_i(kT)}{\Delta u_i(kT)} \right],$$  \hspace{1cm} (11)$$

where $\Delta u_i(\cdot)$ and $\Delta y_i(\cdot)$ are generated from the input and output data.

4. Simulation Examples

Here, in order to examine the effectiveness of the proposed multiple fuzzy controller, let us consider the following equation of motion for the attitude control of a satellite with flexible panels$^{(1,8)}$:

$$\dot{\theta}(t) = 1.764 a(t)$$
$$\ddot{a}(t) = - W^2 a(t) + 4.358 \dot{a}(t)$$
$$y(t) = \theta(t) + 4.358 \dot{a}(t)$$

where $W^2 = 33.15 \times 10^{-4}$ [rad/s$^2$], $\theta(t)$ is the center body rotation due to the rigid body motion, $a(t)$ is the center body rotation due to the flexural motion, $y(t)$ is the measurement of actual attitude, and $u(t)$ is the control input torque produced by reaction jets.

In the following simulations, we assume that the weighting factors $w_i$ are updated at every sampling, and that the sampling width is $T = 0.01$ [s] and one learning trial is made within the control interval 5 [s].

4.1 Simulation 1

In order to check the control performance of a multiple fuzzy controller consisting of two elemental fuzzy controllers, in which both elemental fuzzy controllers have the same number of labels and control rules, we used two elemental fuzzy controllers FC 1 and FC 2, where we used the fuzzy sets consisting of 7 labels as indicated by Fig. 1 and the 49 control rules as tabulated in Table 1. Here, we further defined that

$$x_1 = e(t) = - y(kT)$$
$$x_2 = \dot{e}(t) \approx \Delta e(kT)/T$$

and used the input scalers GIN (1) = 9.0, GIN (2) = 2.1, the output scalers GOU (1) = 0.05 for FC 1, and GIN (1) = 5.0, GIN (2) = 0.01 and GOU (1) = 0.01 for FC 2. The results of a case when each elemental fuzzy controller is singly applied to the plant are shown in Figs. 4 and 5. It is seen from these figures that the control result due to the FC 1 is moderately good, but the one due to the FC 2 is not necessarily satisfactory. However, as seen from Fig. 6, we observe that the result due to the multiple fuzzy controller consisting of these elemental fuzzy controllers gives a satisfac-

![Control result using a single fuzzy controller FC 1](image1)

![Control result using a single fuzzy controller FC 2](image2)

![Control result using a multiple fuzzy controller consisting of FC 1 and FC 2](image3)
Table 2  Control rule table for 5 labels

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>$x_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$NB$</td>
<td>$NM$</td>
</tr>
<tr>
<td>-6</td>
<td>-6</td>
</tr>
<tr>
<td>-3</td>
<td>0</td>
</tr>
<tr>
<td>-6</td>
<td>-3</td>
</tr>
<tr>
<td>-3</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>

Fig. 7  Fuzzy set consisting of 5 labels

Fig. 8  Control result using a single fuzzy controller FC3

multiple fuzzy controller consisting of two elemental fuzzy controllers, in which both elemental fuzzy controllers have different numbers of labels and control rules, we newly introduced an elemental fuzzy controller FC3, under the assumption that the fuzzy set consists of 5 labels as shown in Fig. 7 and 25 control rules as given by Table 1. Note that the control rules of Table 2 were constructed by the following in the same way as used in those of Table 1. Figure 8 depicts the control result for a case when the elemental fuzzy controller FC3 is singly used, where $\text{GIN}(1) = 12.5$, $\text{GIN}(2) = 10.0$ and $\text{GOU}(1) = 0.01$. Figure 9 shows a control result due to a multiple-model fuzzy controller consisting of two elemental fuzzy controllers, FC1, which was used in simulation 1, and FC3, where $\gamma = 0.025$, $\omega_0 = 0.05$, $\omega_1(0) = 0.5$. We obtained a satisfactory response after three learning trials.

4.3  Simulation 3

In order to examine the control performance of a multiple fuzzy controller consisting of three elemental fuzzy controllers, in which all elemental fuzzy controllers have the same number of labels and control rules, we reused the elemental fuzzy controllers FC1 and FC2 that were introduced in simulation 1, under the same condition of fuzzy sets and control rules as used in simulation 2, and defined them as FC1' and FC2', respectively. That is, FC1' and FC2' have the same input and output scalers as those of FC1 and FC2, but with the structure of number of labels 5 and control rules 25. For the multiple fuzzy controller consisting of these FC1', FC2' and FC3, a satisfactory response after six learning trials is depicted in Fig. 10, where $\gamma = 0.02$, $\omega_0(0) = \omega_1(0) = \omega_2(0) = 0.33$.

5.  Conclusions

In this paper we have proposed a multiple fuzzy controller as one of the learning-type fuzzy controllers, in which several fuzzy controllers with
incompletely tuned parameters are processed in parallel and the degree of usage of the inferred consequent from each elemental controller is learned by a linear neural network. The effectiveness of the present approach was shown by simulations for a simple, single input-output system.

The applications of this approach to the trajectory control for a two-link robot manipulator and to the hybrid control of position and force, etc., are now being investigated.

References


