Analysis of a fuzzy logic controller

Citation for published version (APA):

Document status and date:
Published: 01/01/1978

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 25. May. 2019
ANALYSIS OF A FUZZY LOGIC CONTROLLER

W.J.M. KICKERT
Department of Industrial Engineering, Technical University, Eindhoven, P.O. Box 513, Eindhoven, The Netherlands

E.H. MAMDANI
Queen Mary College, Department of Electrical Engineering, London E1 4NS, England

Received May 1977
Revised June 1977

An analysis is performed of the fuzzy logic controller which results in the identity between this controller and a multilevel relay. This tool is used in stability analysis.

1. Introduction

In previous articles a fuzzy logic controller has been proposed [1, 2, 3]. The basic idea behind this approach was to incorporate the "experience" of a human process operator in the design of the controller. From a set of linguistic rules which describe the operator's control strategy a control algorithm is constructed where the words are defined as fuzzy sets. The main advantages of this approach seem to be the possibility of implementing "rule of the thumb" experience, intuition, heuristics and the fact that it does not need a model of the process. This new approach is receiving more and more attention, not only in test cases but also in real industrial applications [4, 5, 6].

An often remarked disadvantage of the method, however, seems to be the lack of appropriate tools for analysis of the controllers performance, such as stability, optimality, etc.

Up till now, apart from some general approaches to fuzzy systems (see e.g. [7] Ch. 4), the main analytical support of this type of controllers is to be found in the investigations of the underlying logical structure [8, 9].

In this paper the stability analysis of the fuzzy logic controllers is treated in two stages. By applying fuzzy logic to the linguistic controller rules it is shown that under fairly general conditions the fuzzy controller is identical to a multilevel relay [10]. This reduces the controller to a conventional nonlinear controller. Herefore we can use the techniques of nonlinear control theory. The particular stability analysis that we will focus upon uses the describing function technique.

2. The fuzzy logic controller

The fuzzy control algorithm used in this research is based on the two concepts of the fuzzy implication and the compositional rule of inference. As this algorithm is clearly
explained in previous reports [1, 2, 3] only a brief formal description will be presented here.

1. The membership function of a fuzzy implication $S$: if $A$ then $B$, given the fuzzy set $A$ of the universe of discourse $X$ and the fuzzy set $B$ of $Y$, is defined by

$$
\mu_A(y, x) = \min[\mu_A(x), \mu_B(y)], \quad x \in X, \quad y \in Y.
$$

\hspace{1cm} (1)

2. Given a fuzzy implication $S$: if $A$ then $B$, the fuzzy set $B'$ inferred by a certain (given) fuzzy set $A'$, where $A$ and $A'$ are fuzzy sets of the universe of discourse $X$, while $B$ and $B'$ are fuzzy sets of $Y$, has a membership function defined by

$$
\mu_{B'}(y) = \max_x \min[\mu_A(x), \mu_A(y, x)], \quad x \in X, \quad y \in Y.
$$

\hspace{1cm} (2)

This is called the compositional rule of inference [11].

Based on these two definitions one can represent in an exact mathematical sense a system which is merely described by a set of linguistic rules, such as

- if “input is big” then “output is medium”
- or (else)
- if “input is medium” then “output is small”
- etc.,

where the or (else) connective is interpreted in the usual sense [11], so that a fuzzy implication $S$ composed of two implications: if $A_1$ then $B_1$, or (else) if $A_2$ then $B_2$, has the membership function

$$
\mu_s(y, x) = \max[\min[\mu_{A_1}(x); \mu_B(y)]; \min[\mu_{A_2}(x); \mu_B(y)]].
$$

\hspace{1cm} (3)

In the considered fuzzy controllers the inputs were measured quantities, hence not fuzzy. Interpreting such an input as a “degenerated” fuzzy set $A'$ with all membership values $\mu_A(x)$ equal to zero, except the value at the measured point $x_0: \mu_A(x_0)$ which is equal to one, the compositional rule of inference reduces to

$$
\mu_{B'}(y) = \max_x \min[\mu_A(x); \mu_B(y)],
$$

$$
= \min[\mu_A(x_0); \mu_B(y, x_0)],
$$

$$
= \mu_s(y, x_0).
$$

\hspace{1cm} (4)

Furthermore, the outputs of the fuzzy controller had to assume deterministic, non-fuzzy values. A decision procedure was to be determined as to decide which particular value had to be chosen from the fuzzy output set of the described algorithm. This point requires some attention because it appeared that the way of analysing a fuzzy controller mainly relies on the particular decision procedure adapted.
There are numerous possibilities for deciding in a fuzzy set $\mu_B(y)$ which particular $y_0 \in Y$ is its good representative. One of the easiest ways is simply to take that value $y_0$ at which the membership function is maximal: i.e. $y_0$ at which

$$\mu_B(y_0) = \max_y \mu_B(y). \tag{5}$$

However, this procedure is not unique when there is more than one maximum membership value. This can be solved by taking the mean of the maxima: i.e. $y_0$ for which

$$y_0 = \frac{1}{J} \sum_{j=1}^{J} y_j, \quad \text{where } \mu_B(y_j) = \max_y \mu_B(y). \tag{6}$$

3. Multilevel relay analogy

A feature which has been noticed in previous research where the controllers had continuous input ranges and a (partial or complete) single-input-single-output structure [3], was that the actual behavior of those fuzzy controllers was similar to that of a multilevel relay when the before mentioned mean of maxima decision procedure was used.

This multilevel relay similarity also becomes apparent when we present the linguistic rules of the controller in the form of a "phase plane" matrix.

In Fig. 1 we represent the rules of the fuzzy logic controller used in [1, 2] in this matrix form: the entries are the different fuzzy input sets for error and change of error, and the cells in the matrix are the corresponding fuzzy output sets for change of plant input. E.g. cell (NS, PS) = NO represents the rule: if "error is NS" then (if "change in error is PS" then "change in action is NO"), where:

- PB: positive big,
- PM: positive medium,
- PS: positive small,
- P0: positive zero,
- (analogously for negative).

![Fig. 1. Matrix representation of fuzzy rules.](image-url)
We see that this is apparently a linguistic multilevel relay: the (linguistic) input area is divided in regions with the same (linguistic) output.

The fuzzy controller however does not only behave as a linguistic multilevel relay, it actually behaves as a real multilevel relay. In case of a single-input-single-output controller the following happens:

The fuzzy sets defined on the input range $X$ induce a partition of that input $X$ into regions (see Fig. 2).

There is a classification of an input into labelled regions according to the fuzzy set where that input belongs; "belongs" has to be interpreted to mean "has the highest membership value". Hence, the points of intersection of neighbouring membership functions are the boundaries between these regions (Fig. 2).

Thus labelled, the fuzzy rule corresponding to that (fuzzy) input label gives the appropriate implied fuzzy output set. Finally it appears that that particular output action is taken where the appropriate fuzzy membership function is maximal (see Fig. 3).

Resuming, the algorithm acts as follows:

The input range is divided into regions and the output range is reduced to points; according to the region where the input falls, one output is chosen. Hence the input-output function of this fuzzy control algorithm is that of Fig. 4. This is obviously a multilevel relay.

We will subsequently show under which conditions this identity between a fuzzy algorithm and a multilevel relay actually holds. The reader is referred to the Appendix for the extensive proof of this identity; here only a short outline of the proof and the results will be presented.
Assume the fuzzy control algorithm consists of $N$ rules $S_i$, $i = 1, 2, \ldots, N$ of the single-input-single-output form: if $A_i$ then $B_i$. The fuzzy sets $A_i$ are defined on the input support set $X$, and the fuzzy sets $B_i$ on the output support set $Y$. Assume furthermore that the action is chosen according to the mean of maxima decision procedure.

So take action

$$y_d = \sum_{j=1}^{J} y_j/J$$

for which

$$\mu_B(y_j) = \max_y \mu_B(y) = \max_y \mu_A(y, x_0)$$

$$= \max \max_i \min_y \left[ \mu_{A_i}(x_0); \mu_{B_i}(y) \right] \quad i = 1, 2, \ldots, N.$$  \hspace{1cm} (7)

Assume that:

(a) all fuzzy output sets are normal
which means that their membership functions attain the value one for at least one $y$. Now eq. 7 can be proved to be identical to

$$\max_{y} \min_{i} \left[ \mu_{A_{i_0}}(x_0); \mu_{B_{i_0}}(y) \right],$$  \hspace{1cm} (8)

where $i_0$ is determined from $\mu_{A_{i_0}}(x_0) = \max_i \mu_{A_i}(x_0)$.

This latter formula (8) implies that one first decides to which input set $A_{i_0}$ the input $x_0$ "belongs" and only works out the corresponding rule (rule number $i_0$). By examining the resulting fuzzy output set $B_{i_0}$, it can be shown that the final computed control action $y_d$ will be the point at which

$$\mu_{B_{i_0}}(y_d) = \max_y \mu_{B_{i_0}}(y)$$

when one of the following (sufficient) conditions is satisfied:

(b$_1$) $\mu_{A_{i_0}}(x_0) = 1$,

(b$_2$) $\mu_{B_{i_0}}(y)$ is symmetrical around its maximum $y_d$.  

Some additional facts can be shown as well:

—when \( \max_i \mu_{A_i}(x_0) = \mu_{A_{i_0}}(x_0) \) for several different \( i_0 \) the final output \( y_d \) will be the mean of the several corresponding maxima \( y_d \).

—when a fuzzy output set is not symmetrical, the algorithm remains a multilevel relay in all other regions.

Note that the fact that the input lies at a unity membership function point is highly improbable in the continuous case and can easily be made impossible in the discrete case.

3.1. Multiple inputs

The fuzzy control algorithms which have been implemented in [1, 2, 3] had a two-input-single-output structure with rules of the form: if \( A \) then (if \( B \) then \( C \)), so that the algorithm is defined as follows (compare with eq. 7). Take action

\[
y_d = \sum_{j=1}^{J} y_j/J
\]

for which

\[
\mu_B(y_j) = \max_y \mu_B(y) = \max_y \max_i \min[\mu_{A_i}(x_0) \cdot \mu_{B_i}(dx_0) \cdot \mu_{C_i}(y)].
\]

(9)

It can be shown that in this two-input case the foregoing identity with a multilevel relay—now two dimensional—still holds under the conditions (see Appendix):

(a) all fuzzy output sets \( C_i \) are normal: \( \max_y \mu_{C_i}(y) = 1 \) and one of the two conditions:

(b1) \( \mu_{A_{i_0}}(x_0) = 1 \) and \( \mu_{B_{i_0}}(dx_0) = 1 \),

(b2) The fuzzy output membership function \( \mu_{C_{i_0}}(y) \) is symmetrical around its maximum,

with the extra condition for the two-input case

(c) The algorithm consists of an exhaustive set of linguistic rules; there is a rule for every possible combination of fuzzy input sets \( (A_i \text{ and } B_j) \).

The extension of these conditions to a multiple input case is straightforward, as can be seen from the proof in the Appendix.

4. Properties of a multilevel relay: Describing function

One of the tools for analyzing nonlinear systems, especially sustained oscillations, is the describing function technique [12]. We assume that the multilevel relay is
symmetrical with respect to zero. We can subdivide this multilevel relay element into the nonlinear elements of Fig. 5, and one final relay with dead zone $x_N$.

![Fig. 5. Nonlinear element.](image)

The elements have a describing function

$$DF = \frac{4y_i}{\pi x^2} \left[ \sqrt{x^2 - x_i^2} - \sqrt{x^2 - x_{i+1}^2} \right], \quad (10)$$

(as can easily be obtained by subtracting the $DF$'s of two relays with dead zone) and the final relay with dead zone for the last level $y_N$ has a describing function

$$DF = \frac{4y_N}{\pi x^2} \sqrt{x^2 - x_N^2}. \quad (11)$$

Hence a general multilevel relay with $N$ input levels $x_1 \ldots x_N$ and $N$ corresponding output levels $y_1 \ldots y_N$ has a describing function defined by

$$DF = \frac{4}{\pi x^2} \left\{ \sum_{i=1}^{N-1} y_i \left[ \sqrt{x^2 - x_i^2} - \sqrt{x^2 - x_{i+1}^2} \right] + y_N \sqrt{x^2 - x_N^2} \right\}. \quad (12)$$

In the case no dead zone exists, take $x_1 = 0$. This describing function has a configuration as shown in Fig. 6.

![Fig. 6. Describing function multilevel relay.](image)
When the quantization levels \( x_i \) lie very near to each other we get the DF configuration of Fig. 7, and when the levels \( x_i \) lie far from each other we get Fig. 8.

**Fig. 7.** \( X_1 \approx X_2 \approx X_3 \).

**Fig. 8.** \( X_1 \ll X_2 \ll X_3 \).

5. The Prediction of oscillations

The describing function of this nonlinear element can now be used to analyse the existence of autonomous oscillations in a feedback system which this element is part of. Assuming that the transfer function \( H(j\omega) \) of the linear part of the system acts as a low pass filter, the input to the element can be considered to be sinusoidal, hence the DF technique applies. Then a sufficient condition for the existence of sustained oscillations in this feedback system is: \( H(j\omega) = -\frac{1}{\text{DF}} \).

In this case study the linear part of the system was a dead time system with one time constant.

\[
H(j\omega) = K \frac{\exp(-j\omega T_d)}{(j\omega + a)}. \tag{13}
\]

Because of the absence of an imaginary part in the DF of the multilevel relay, the condition for oscillation will always occur at \( \omega = T_d \). For this particular system this condition can easily be calculated to occur when

\[
tg(\omega T_d) = -\frac{\omega}{a}. \tag{14}
\]

This equation can be displayed graphically by plotting \( tg(\omega T_d) \) and \( \omega/a \) in one graph (Fig. 9).

It is clear that when \( a \) is small (say \( 1/a \gg T_d \)) the condition can be approximated by

\[
\omega = (k + \frac{1}{2}) \frac{\pi}{T_d}, \quad k = 0, 1, 2, \ldots, \tag{15}
\]
Fig. 9. Graphical solution of \( t g \omega T_d = -\frac{\omega}{\omega} \).

Then the system gain will become

\[
H(j\omega) = -\frac{\omega K}{a^2 + \omega^2}.
\]

(16)

An experiment was carried out to verify whether oscillations of a fuzzy controller/multilevel relay could be predicted. The above mentioned linear system was simulated using fourth order Runge–Kutta integration.

Time constant \( a = 0.05 \text{ sec}^{-1} \),

system gain \( K = 0.05 \),

time delay \( T_d = 12 \text{ sec} \),

hence the (lowest) frequency of oscillation will be

\[
\omega = \frac{\pi}{24} \quad (48 \text{ sec cycle period}),
\]

(note that indeed \( 1/a \gg Td \)). The system gain is then

\[
H(j\omega) = 0.34.
\]

Assuming that the input levels of the relay lie far from each other, the maximal \( DF \) can be approximated as follows (see Fig. 8)

first level \( DF = \frac{2}{\pi} \frac{y_1}{x_1} \),

second level \( DF = \frac{2}{\pi} \frac{y_2 - y_1}{x_2} \),

third level \( DF = \frac{2}{\pi} \frac{y_3 - y_2}{x_3} \).
An oscillation should theoretically occur when $H(j\omega) = -1/DF$.

The three multilevel relays which were implemented had a configuration of quantization levels as summarized in Table 1 where the maximal $DF$ has been filled in at the level where it occurs. The results of the oscillation experiments are summarized in Table 2.

Table 1
Multilevel relay configurations

<table>
<thead>
<tr>
<th>Input levels</th>
<th>Output levels</th>
<th>Maximal DF</th>
</tr>
</thead>
<tbody>
<tr>
<td>First type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4.4</td>
<td>2.78</td>
</tr>
<tr>
<td>9</td>
<td>30.8</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>35.2</td>
<td></td>
</tr>
<tr>
<td>Second type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.9</td>
<td>2.17</td>
</tr>
<tr>
<td>4</td>
<td>16.4</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>28.7</td>
<td></td>
</tr>
<tr>
<td>Third type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.9</td>
<td>2.63</td>
</tr>
<tr>
<td>2</td>
<td>6.2</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>41.0</td>
<td></td>
</tr>
</tbody>
</table>

Table 2
Results of test case

<table>
<thead>
<tr>
<th></th>
<th>First type</th>
<th></th>
<th>Second type</th>
<th></th>
<th>Third type</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Theory</td>
<td>Practice</td>
<td>Theory</td>
<td>Practice</td>
<td>Theory</td>
<td>Practice</td>
</tr>
<tr>
<td>$DF^{-1}$</td>
<td>0.34</td>
<td>0.36</td>
<td>0.34</td>
<td>0.46</td>
<td>0.34</td>
<td>0.38</td>
</tr>
<tr>
<td>ampl.</td>
<td>1.4</td>
<td>1.4</td>
<td>5.6</td>
<td>5.4</td>
<td>14.1</td>
<td>13.5</td>
</tr>
<tr>
<td>period</td>
<td>48</td>
<td>44</td>
<td>48</td>
<td>44</td>
<td>48</td>
<td>44</td>
</tr>
</tbody>
</table>

The differences between the theoretically predicted results and the actually measured results can partially be accounted for by the assumed approximations, but this does not seem meaningful since the $DF$ method is itself an approximation, especially in this test case with only a first order system.

Further decrease of the input-output level ratio resulted in oscillations of higher amplitude but with identical frequency, which matches with theory [12]. Increase of level ratio suppressed the oscillations.

These facts affirm the theorem that in general a $N$-level relay (with dead zone) will enable $2N$ possible oscillations, of which $N$ cases are unstable (lower amplitude) and $N$ stable (higher amplitude). Obviously a sufficient (but too strong) condition to avoid oscillations is to choose $y_i$ and $x_i$ such that for all $i$, $2y_i/(\pi x_i)$ is smaller than $|H(j\omega)|$ at $\angle H(j\omega) = -\pi$. In case there is no dead zone $2N - 1$ oscillations are possible, $N$ stable.
and \( N - 1 \) unstable and at least one oscillation (the lowest amplitude) can never be avoided.

6. Discussion

The identity between the fuzzy algorithm and the multilevel relay that we have proven, is rather general.

The particular nonlinear control technique that we used—the describing function technique—is however rather restricted. At first the technique relies on the assumption of a low-pass linear system, hence sinusoidal inputs. A first order system is not a very good low pass filter. (This might account for the fairly large differences between the characteristics of the predicted and the actual oscillations.)

Secondly the describing function technique as used above, only applies to the single-input case. Deriving describing functions for a multidimensional multilevel relay is generally speaking not so meaningful.

Luckily the describing function technique is only one of the tools available to investigate nonlinear systems. Another well known method is the state space approach. By constructing state trajectories a general view of the system can be obtained. The extension of the phase plane investigations of a relay with dead zone is straightforward. Exactly as in the case of a relay with dead zone, possible steady state errors can be explained by constructing the trajectories in the phase plane by means of the isocline method (see e.g. [13] p. 274). However, the phase plane method restricts the investigatable linear parts of the system to second order, hence a stability investigation is not applicable.

Because the nonlinear element—the fuzzy algorithm—cannot be described by an analytical function, most of the modern nonlinear system theory is not applicable. Only in the single-input-single-output case could methods like the stability criterion of Popov be used as well.

Generally speaking the analysis of fuzzy controllers by means of nonlinear control theory can never be general, simply because a general theory of nonlinear control systems does not exist.

If we look for generality a possible alternative way of analyzing fuzzy controllers might be to start from Boolean or lattice algebra especially matrix algebra. In the case of discrete fuzzy sets, the controller namely comes down to a fuzzy relation matrix.

Some examples of this type of approach can be found in [14, 15]. A less abstract investigation of the fuzzy controller relation matrix is performed in [16]. However, these studies have not yet resulted in conclusions about typical control properties such as stability.

7. Conclusion

Using the mean of maxima decision procedure at the end of the fuzzy control algorithm it has been shown that the controller is identical to a multilevel relay under certain conditions. This is done by consequently applying the principles of fuzzy logic to the linguistic rule structure of the fuzzy controller.
Given the plant model the way is now opened to investigate the nonlinear control system in an analytical way. One particular analytical technique from nonlinear control theory has been investigated more thoroughly, namely the describing function technique. With this tool a frequency domain stability analysis has been carried out. The theoretical results of this analysis fitted the practical results. Because the describing function technique puts some restrictions to the order of the system and the dimensionality of the inputs, several other methods from nonlinear control theory have also been discussed. Moreover another more general approach to the analysis of fuzzy logic control has been discussed.

8. Appendix

The fuzzy control algorithm is defined by:

Take action

\[ y_d = \sum_{j=1}^{J} y_j / J \]

for which

\[ \mu_{B_i}(y_j) = \max_y \max_i \min \mu_{A_i}(x_0); \mu_{B_i}(y) \], \quad i = 1, 2, \ldots, N. \quad (A1) \]

Now assume that all fuzzy output sets \( B_i \) are normal, which means that

\[ \forall i, \exists y : \max_y \mu_{B_i}(y) = 1. \]

For a normal fuzzy membership function

\[ \max_y \min[a; \mu(y)] = a, \quad a \in [0, 1]. \quad (A2) \]

Hence

\[ \max_y \max_i \min[a; \mu_{A_i}(x_0); \mu_{B_i}(y)] = \max_i \max_y \min[a; \mu_{A_i}(x_0); \mu_{B_i}(y)] \]

\[ = \max_i \mu_{A_i}(x_0) = \mu_{A_{i_0}}(x_0), \quad (A3) \]

where the first equality reflects the commutativity of the maximum operator. In the same way it can be shown that

\[ \mu_{B_i}(y_j) = \max_y \min[a; \mu_{A_i}(x_0); \mu_{B_{i_0}}(y)] \]

\[ = \max_i \mu_{A_i}(x_0) = \mu_{A_{i_0}}(x_0). \quad (A4) \]
In the algorithm represented by eq. (A4) one first decides to which fuzzy input set \( A_{i_0} \) the input \( x_0 \) "belongs" and only works out the corresponding rule (rule number \( i_0 \)). Notice however that the equality of (A1) and (A4) does not yet permit the conclusion that the control actions \( y_d \) are equal in both cases. Herefore we have to examine the fuzzy output set:

\[
\mu_B(y) = \max_i \min \left[ \mu_{A_i}(x_0); \mu_B(y) \right]
\]  

(A5)
in greater detail.

In case of three rules this fuzzy output set will have a configuration as shown in Fig. 10 (thick line).

The action obtained by taking the maximum over this fuzzy set is equal to the range \( y_{2 \text{max}} \). Identically the computation of \( y_j \) from the alternative algorithm:

\[
\mu_{B'}(y_j) = \max \min \left[ \max_i \mu_{A_i}(x_0); \mu_{B_{i_0}}(y) \right]
\]  

(A6)
will give \( y_{2 \text{max}} \) because \( \mu_{A_2}(x_0) = \max_i \mu_{A_i}(x_0) \).

Bearing Fig. 10 in mind it can be imagined that when there is more than one \( i_0 \) for which

\[
\mu_{A_{i_0}}(x_0) = \max_i \mu_{A_i}(x_0),
\]

the action will be the same in both cases, namely several separate ranges \( y_{i_0 \text{max}} \). This situation will occur when the input \( x_0 \) happens to be at a point of intersection of neighbouring fuzzy input sets. Hence, in general, in both algorithms that range of control action \( y_j \) will result for which

\[
\mu_{B'}(y_j) = \mu_{A_{i_0}}(x_0) = \max_i \mu_{A_i}(x_0),
\]  

(A7)
which corresponds to the fuzzy output set

\[
\mu_{B'}(y) = \min_i \left[ \mu_{A_{i_0}}(x_0); \mu_{B_i}(y) \right],
\]  

(A8)
because of the identity of eqs. (A3) and (A4).
This is the proof of identity between the algorithms of eqs. (A1) and (A4).

Still bearing Fig. 10 in mind it can be observed that in case the mean of maxima decision procedure is applied to the computed range of control action \( y_p \), the final control action will be the point \( y_d \) where

\[
\mu_{B_0}(y_d) = \max_y \mu_{B_0}(y)
\]

(namely the mean of the maximum range \( y_j \)) when one of the following (sufficient) conditions is satisfied:

(a) \( \mu_{A_0}(x_0) = 1 \),

(b) \( \mu_{B_0}(y) \) is symmetrical around its maximum \( y_d \).

**Multiple inputs**

The two input fuzzy algorithm is defined by:

Take action

\[
y_d = \sum_{j=1}^{\infty} \frac{y_j}{J},
\]

for which

\[
\mu_B(y_j) = \max_y \max_i \min[\mu_{A_i}(x_0); \mu_{B_i}(d_{x_0}); \mu_{C_i}(y)]. \tag{A9}
\]

Analogously to the previous single input case the identity with a (two dimensional) multilevel relay can be proven to hold when eq. (A9) can be proven to be identical to

\[
\mu_B(y_j) = \max_y \min_i \max_j [\mu_{A_i}(x_0); \mu_{B_j}(d_{x_0}); \mu_{C_{i,j}}(y)]. \tag{A10}
\]

By using the properties of associativity and distributivity it can be shown that

\[
\min_i \max_j [\mu_{A_i}(x_0); \mu_{B_j}(d_{x_0})] = \max_i \min_j [\mu_{A_i}(x_0); \mu_{B_j}(d_{x_0})]. \tag{A11}
\]

Hence the condition under which (A10) is equal to (A9) is that the maximum over all rules (rule number \( i \)) is equal to the maximum over all fuzzy input sets (input numbers \( i \) and \( j \)). In other words:

(c) The fuzzy control algorithm has to consist of an exhaustive set of linguistic rules; there should be a rule for every combination of fuzzy input sets \( A_i \) and \( B_j \).
The previously derived conditions are slightly changed into

(a) \( \max A_i(x_0) = 1 \) and \( \max B_j(dx_0) = 1 \)

(b) \( \mu_{C_{x_0}}(y) \) is symmetrical around its maximum.

It should be remarked that the merging of rules like "if \( A \), then \( B \)" or (else) "if \( A_2 \) then \( B \)" into the form "if \( A_1 \) or \( A_2 \) then \( B \)" is of no influence to condition (c) since

\[
\min[\max[\mu_{A_1}(x_0); \mu_{A_2}(x_0)]; \mu_B(y)] \\
= \max[\min[\mu_{A_1}(x_0); \mu_B(y)]; \min[\mu_{A_2}(x_0); \mu_B(y)]]
\]

because this is precisely the rule of distributivity

\[
(a \lor b) \land c = (a \land c) \lor (b \land c).
\]

Acknowledgement

We wish to thank Professor H.R. van Nauta Lemke for enabling the first author to perform the case study of Section 5 at his laboratory at Delft Technological University.

References
