Appendix A Fuzzy logic and fuzzy control

A.1 Introduction

In this appendix a brief introduction is given about the basics of the *fuzzy set theory* and *fuzzy control*. To understand this report, one has to be abreast of the basics of fuzzy logic and fuzzy control. However, many introductions similar to this one have been published over the years in reports, articles and books about fuzzy control. Most readers of this report will probably have read some of those introductions and therefore this part about fuzzy logic and fuzzy control is put in an appendix and may still be interesting to brush up one's knowledge on this subject.

In 1965 *Zadeh* [1965] wrote a paper in which he introduced fuzzy sets, sets with *unsharp* boundaries.These would be in better agreement with the human mind, that works with shades of grey, rather than with just black or white. Typical for fuzzy sets is that they are able to represent linguistic terms, e.g. warm, hot, high, low. Nearly 10 years later Mamdani [1974] succeeded to apply fuzzy logic for control in practice. By now, in Japan fuzzy control is widely accepted and applied. In many electronical consumer product, like washing-machines and cameras, fuzzy controllers are used in order to obtain more intelligent and userfriendly products. A few interesting applications can be mentioned: control of subway systems, image stabilisation of video cameras and autonomous control of helicopters. The United States and Europe still hesitate accepting fuzzy control, since initially stability could not be proved. However, recently many papers on different stability analysis methods have appeared. For more detailed information about these stability methods is referred to Section 4.5.

In Section A.2 the differences between fuzzy sets and classical crisp sets will be discussed. The next section (A.3) deals with the operations on fuzzy sets that are used for fuzzy control. In Section A.4 the *Fuzzy Logic Controller* (FLC) is explained. Notice that this introduction is not complete, but just provides the basic knowledge that is indispensable for understanding this report.

A.2 Fuzzy sets versus crisp sets

In the classical set theory, a set is denoted as a so-called *crisp set* and can be described by its characteristic function as follows:

$$\mu_C\colon U \Rightarrow \{0, 1\} \tag{A.1}$$

In equation (A.1) U is the *universe of discourse*, i.e. a collection of elements that can be continuous or discrete. In a crisp set each element of the universe of discourse either belongs to the crisp set ($\mu_C = 1$) or does not belong to the crisp set ($\mu_C = 0$).

Consider a characteristic function μ_{Chot} representing the crisp set *hot*, a set with all 'hot' temperatures. Figure A.1 graphically describes this crisp set, considering temperatures higher than 25 degrees as hot. (Note that for all the temperatures t hold: $t \in U$)

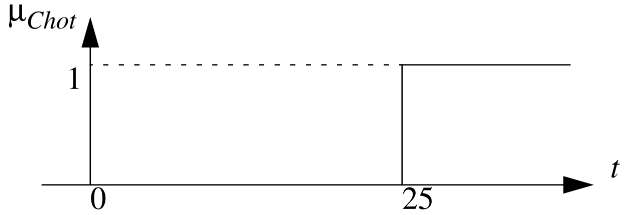

Figure A.1: The characteristic function μ_{Chot}

The definition of a *fuzzy set* [Zadeh, 1965] is given by the characteristic function

$$\mu_F\colon U \Rightarrow [0,1] \tag{A.2}$$

In this case the elements of the universe of discourse can belong to the fuzzy set with any value between 0 and 1. This value is called the *degree of membership*. If an element has a value close to 1, the degree of membership, or 'truth' value is high. The characteristic function of a fuzzy set is called the *membership function*, for it gives the degree of membership for each element of the universe of discourse. If now the characteristic function of μ_{Fhot} is considered, one can express the human opinion, being for example that 24 degrees is still fairly hot, and that 26 degrees is hot, but not as hot as 30 degrees and higher. This results in a gradual transition from membership (completely true) to non-membership (not true at all). In Figure A.2 is shown the membership function μ_{Fhot} of the fuzzy set F_{hot}.

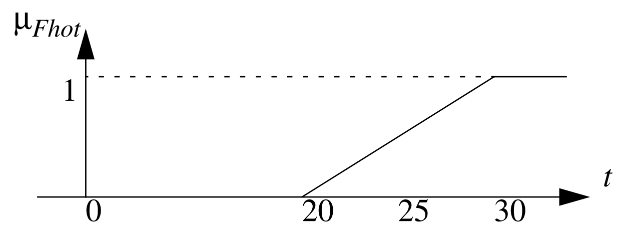

Figure A.2: The membership function μ_{Fhot}

In the figure above the membership function has a linear transition. However, every individual can construct a different transition according to his own opinion. Membership functions can have any possible shape. In practice the transitions are often linearized to simplify the computations.

A.3 Fuzzy logic

Just like to traditional crisp sets, logical operations, e.g. union, intersection and complement, can be applied to fuzzy sets [Zadeh, 1965]. Since some of these operations are used for fuzzy control, the necessary operations are discussed is this section.

Union: The union operation (and the intersection operation as well) can be defined in many different ways. Here, only discussed is the definition that is used in most cases. The union of two fuzzy sets A and B with the membership functions $\mu_A(x)$ and $\mu_B(x)$ is a fuzzy set C, written as $C = A \cup B$, whose membership function is related to those of A and B as follows:

$$\forall x \in U: \mu_C(x) = \max\,[\mu_A(x), \mu_B(x)] \tag{A.3}$$

where U is the universe of discourse. The operator in this equation is referred to as the *max-operator.*

Intersection: According to the *min-operator*, the intersection of two fuzzy sets A and B with respectively the membership functions $\mu_A(x)$ and $\mu_B(x)$ is a fuzzy set C, written as $C = A \cap B$, whose membership function is related to those of A and B as follows:

$$\forall x \in U: \mu_C(x) = \min\,[\mu_A(x), \mu_B(x)] \tag{A.4}$$

Both the intersection and the union operation are explained by Figure A.3.

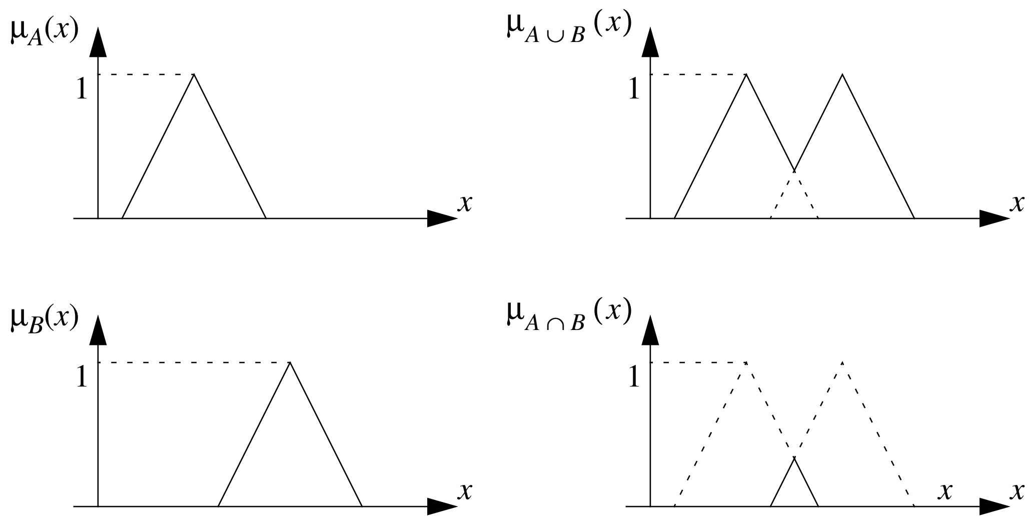

Figure A.3: The fuzzy set operations union and intersection

Complement: The complement of a fuzzy set A is denoted $\overline{A}$ as with a membership function defined as (see also Figure A.4):

$$\forall x \in U : \mu_{\overline{A}}(x) = 1 - \mu_A(x) \tag{A.5}$$

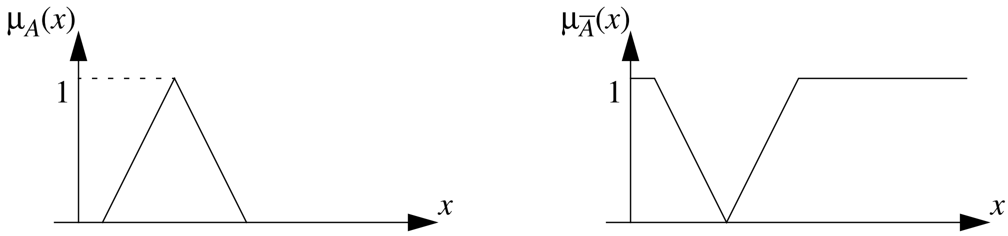

Figure A.4: The intersection operation

Most of the properties that hold for classical sets (e.g. commutativity, associativity and idempotence), hold also for fuzzy sets, manipulated by the specific operations in equations (A.3)-(A.5), except for two properties:

1. *Law of contradiction* ($A \cap \overline{A} = \varnothing$): One can easily notice that the intersection of a fuzzy set and its complement results in a fuzzy sets with membership values up to 1/2 and thus not equals the empty set (see the equation below)

$$\forall x \in U : \mu_{A \cap \overline{A}}(x) = \min[\mu_A(x), (1 - \mu_A(x))] \le \frac{1}{2} \ne \varnothing \tag{A.6}$$

2. *Law of excluded middle* ($A \cup \overline{A} = U$): The union of a fuzzy set and its complement does not give the universe of discourse:

$$\forall x \in U : \mu_{A \cup \overline{A}}(x) = \max[\mu_A(x), (1 - \mu_A(x))] \ge \frac{1}{2} \ne U \tag{A.7}$$

A.4 Fuzzy control

The principal idea of fuzzy control is control based on linguistic terms. By using '*IF* .. *THEN* ..'-rules, the linguistic terms of the input variable(s) generate a linguistic control signal. Suppose that a heather is controlled by linguistic '*IF* .. *THEN* .. '-rules, two of these rules could be:

IF temperature is hot THEN turn heather off

IF temperature is cold THEN turn heather on

where the elements of the *IF*-part are called the *antecedents* and the elements of the *THEN*-part are called the *consequents*. This example points out that a fuzzy controller is able to represent human thinking and reasoning. Where for conventional control a mathematical model of a system is indispensable, for fuzzy control experience of an expert is sufficient to obtain the fuzzy rules. The set of all the fuzzy rules is called the *Fuzzy Associate Memory* (FAM) or simply *rule base*.

Furthermore, each linguistic term has to be defined in a numerical way by a characteristic function. An example can be found in A.2, where the term 'hot' is defined as a membership function. For this stage precise information about the terms is necessary. Often, simple shapes were used to represent a linguistic term, such as triangular and trapezoidal shapes, however if one uses a design tool (see Chapter 2) one is usually allowed to define the membership function in any desired shape.

In may be clear that, for example, a temperature sensor does not produce a fuzzy term, but a crisp value. Therefore so-called *fuzzification* is applied to the sensor signals, to *fuzzify* the crisp values into linguistic terms, and likewise *defuzzification* is needed to obtain the crisp control value. The rule base and, additional to this, the information necessary for proper functioning of the fuzzification module, the rule base and the defuzzification module is called the *knowledge base*. Therefore, a FLC is sometimes referred to as a *Fuzzy Knowledge Based Controller* (FKBC).

Usually one works with a normalized universe of discourse (for example: U = [-1, 1]) for the input and output variables, so *normalisation* and *denormalisation* is needed.

In Figure A.5 the different elements of a FLC, of which most of are them mentioned above, are represented in a block diagram [Olafsson *et al.,* 1995]. The *inference engine* module combines the fuzzified input variables and the rule base and calculates the fuzzy output.

In Sub-section A.4.1 some additional information is supplied about membership functions and rules, in order to allow one to understand the remaining part of this section. In the following sub-sections (A.4.2-A.4.4) the different elements of a FLC are discussed. The FLC as described in these sub-sections is usually referred to as the *Mamdani*-type fuzzy controller. In Sub-section A.4.5 a special fuzzy controller type, the *Takagi-Sugeno*-type, is explained, for this one is mostly used for software implementation because of its small execution time. Finally (see A.4.6), some advantages and disadvantages of fuzzy logic are summarized.

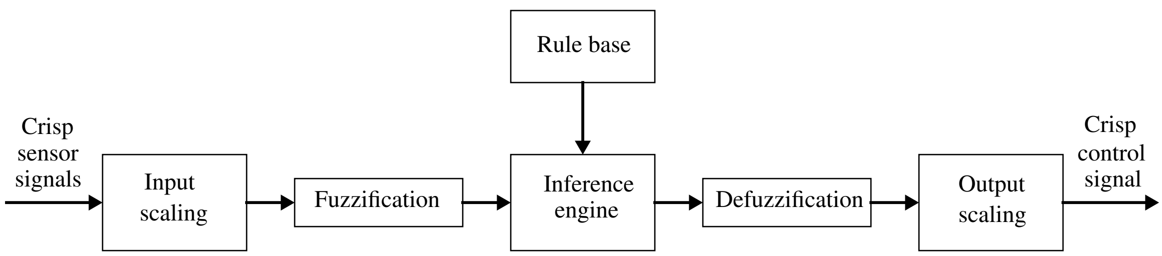

Figure A.5: General block diagram of a fuzzy controller

A.4.1 Membership functions and rules

Before the a FLC can operate, both the membership functions of the variables rules have to be defined.

For example, for the variable ‘temperature’ one can not just define the membership function ‘hot’, but should also define terms like ‘cold’ and ‘comfortable’. The whole chosen universe of discourse must be supported by membership functions. In most control systems, the input variables of a FLC are error signals, their derivatives or their integral. These variables can be supported by terms like Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive small (PS), Positive Medium (PM), and Positive Big (PB). In this introduction is just discussed the most basic (but in fact mostly used) way to define the membership functions of the above mentioned terms (see Figure A.6). In Figure A.6 triangular shapes are used for all terms and these are equally distributed over the universe of discourse. Using other membership function shapes can change the behaviour of the FLC considerable. Furthermore, the width of the triangles and the way they are distributed are of great influence as well. This aspect will, however, not be discussed in this introduction and for more information is referred to Driankov *et al.* [1993].

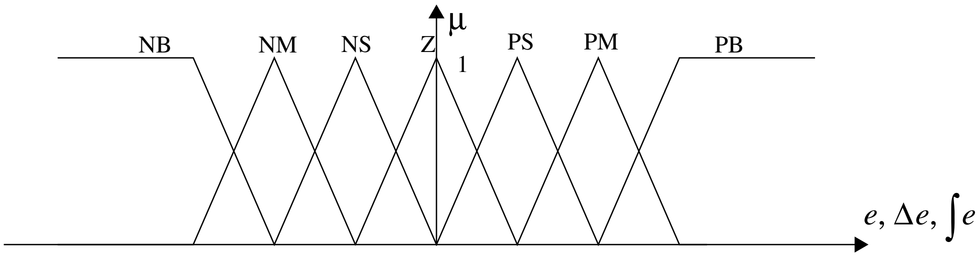

Figure A.6: Example of a complete term set

If all variables are defined in terms of NB, NM, etc. the antecedents and consequents can be connected by rules, meaning that one can develop the rule-base. In order to be able to control every combination of sensory values, one rule has to defined for *all* possible combinations of membership functions of the input variables. Consider a FLC with 2 input variables and for each variable 7 membership functions like mentioned above. That leads to a number of rules of $7^2 = 49$. Notice that the number of rules grows exponentially with the number of inputs. Chapter 3 of this report deals with this problem. This sub-section considers only the 2 input case. Note that when dealing with fuzzy control one can always determine the rules by reasoning.

Now consider a FLC for a PD-like control problem with the inputs e and Δe and an output u. One rule could have the form:

IF e is Z and Δe *is NS THEN u is PS*

This rule deals with the situation in which the error is about zero but the deviation of the error NS, meaning that the error will get negative. Just by reasoning one can conclude that the control signal should be positive to prevent the error signal from getting negative. Furthermore, a small correction should be enough, thus the control signal should be PS. In this way all rules can be reasoned. The complete rule base of a 2-input FLC can be represented graphically. The graphical representation of the PD-like FLC can be found in Figure A.7. This rule base is frequently used for PD-like and PI-like control problems and is called a linear McVicar-Whelan-type rule-base.

Δe \ e	NB	NM	NS	Z	PS	PM	PB
PB	Z	NS	NM	NB	NB	NB	NB
PM	PS	Z	NS	NM	NB	NB	NB
PS	PM	PS	Z	NS	NM	NB	NB
Z	PB	PM	PS	Z	NS	NM	NB
NS	PB	PB	PM	PS	Z	NS	NM
NM	PB	PB	PB	PM	PS	Z	NS
NB	PB	PB	PB	PB	PM	PS	Z

Figure A.7: McVicar-Whelan-type rule-base

A.4.2 Normalisation and denormalisation

Normalisation, or scaling, of the crisp input values is not strictly necessary. One can define the membership functions for the input variables on their physical domain [Driankov *et al.*, 1993]. In that case scaling is superfluous.

However, mostly both inputs and outputs of the controller are defined on a common domain, meaning the same universe of discourse (for example: U = [-1, 1]). The physical values are being mapped on the predetermined domain by multiplying the physical input value with a *normalisation factor* as follows:

$$x_{norm} = N_x \cdot x \tag{A.8}$$

where x is the physical input value, x_{norm} the normalized value (so mapped on the universe of discourse) and N_x the normalisation factor. The advantage of this approach is that the fuzzification, inference engine and defuzzifications modules can be designed independently of the physical domains. Furthermore, the normalisation factor is a tuning parameter of the FLC (se ..)

Likewise, denormalisation is not necessary. However, in case the domain of the output variable is normalized, denormalisation is applied as follows:

$$y_{denorm} = N_y \cdot y \tag{A.9}$$

where y is the normalised output value, y_{crisp} is the physical value and N_y the denormalisation factor.

A.4.3 Fuzzification and inference

The fuzzification and inference phase are difficult to separate. In this sub-section the two most well known approaches, the *max-min* and the *max-dot* approach, will be discussed. [Nauta

Lemke, 1993, Thomson, 1994a, Titli, 1995]

The purpose of this phase is to fuzzify the crisp input and then to check which rules of the rule-base have a 'truth' value above zero. The higher the 'truth' value of a rule, the more the final output signal will be influenced by this rule. This phase results in a fuzzy output signal. This phase is best explained by Figures, therefore first an example with Figures is given and next the general formulas for the max-min and max-dot approach are derived.

Consider the PD-like FLC from Section A.4.1. Suppose at a certain moment both e and Δe have a crisp value between the centres of the fuzzy sets Z and PS. In the particular case of the FLC described in Section A.4.1 4 rules will be *fired*, meaning, will have a 'truth' value above zero. In order to provide a clear explanation of the fuzzification and inference phase, only the following 2 of the 4 rules will be take into account:

IF e is Z and Δe *is Z THEN u is Z*

IF e is Z and Δe *is PS THEN u is NS*

For these two rules the fuzzification is graphically explained by Figure A.7 and A.8 using respectively the max-min and max-dot approach. First, for each rule, the minimun is taken of the antecedents membership values related with the crisp inputs. Then, the consequents membership function is either cut off above this minimum ('clipped') or multiplied by this minimum ('scaled'), depending on what approach is used. The last step builds up a combined fuzzy output set by calculating the union of all rules results, for the here discussed methods, by using the max-operator (see Section A.2).

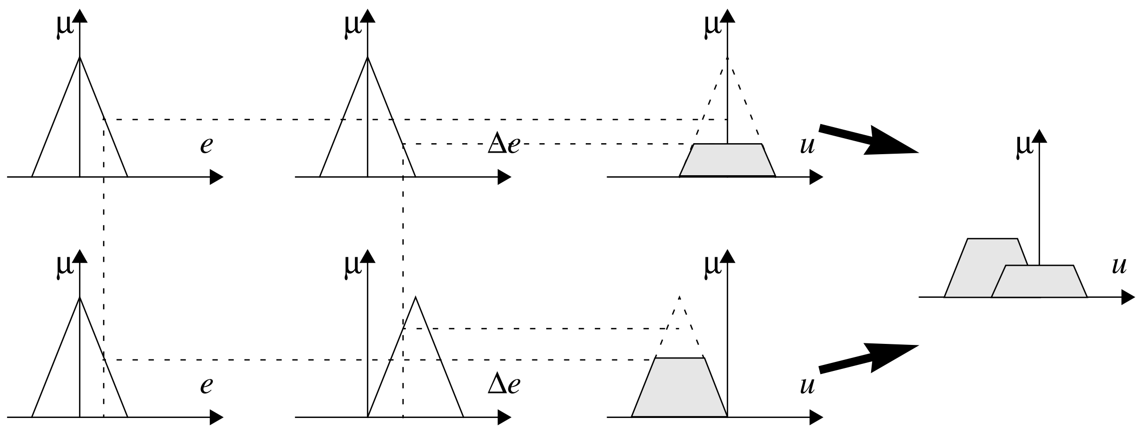

Figure A.8: An example of the fuzzification and inference phase according to max-min

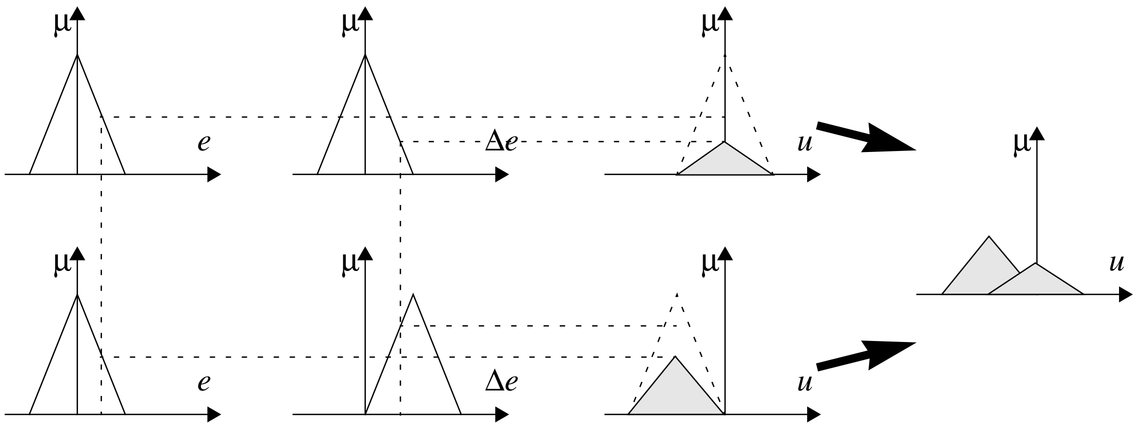

Figure A.9: An example of the fuzzification and inference phase according to max-dot

After this example the general equations are derived. Consider a r rules FLC of which the j^{th} rule has the form

$$R_j\text{: } IF\ x_1\ is\ A_{j1}\ and\ x_2\ is\ A_{j2}\ and\ ...\ and\ is\ A_{jn}\ THEN\ y\ is\ B_j \tag{A.10}$$

where $x_1 ... x_n$ are the crisp inputs, y is the output variable and $A_{j1} ... A_{jn}$ and B_{j1} are fuzzy sets of any shape. If U is the universe of discourse of the output, the following equation determines the outputs membership function $\mu_{Output}(y)$ according to the max-min approach:

$$\forall y \in U: \quad \mu_{Output}(y) = \bigcup_{j=1}^{r} \{\min(\mu_{A_{j1}}(x_1), \mu_{A_{j2}}(x_2), \ldots, \mu_{A_{jn}}(x_n), \mu_{B_j}(y))\} \tag{A.11}$$

For the max-dot approach the following equation can be obtained:

$$\forall y \in U: \quad \mu_{Output}(y) = \bigcup_{j=1}^{r} \{\min(\mu_{A_{j1}}(x_1), \mu_{A_{j2}}(x_2), \ldots, \mu_{A_{jn}}(x_n)) \cdot \mu_{B_j}(y)\} \tag{A.12}$$

Notice that for the union operation in both equations above the max-operator has to be used. The next section deals with obtaining a crisp output from the outputs membership function $\mu_{Output}(y)$.

A.4.4 Defuzzification methods

In this section is explained how a crisp output value can be extracted from the fuzzy output set, according to the well-known 'Centre Of Gravity' (COF) and 'Centre of Sums' (COS) methods. Over the years many more methods have been proposed, and for an extensive comparison of a large set of defuzzification methods is referred to Driankov *et al.* [1993].

Centre of Gravity:

This method determines the centre of gravity of the area below the combined membership function $\mu_{Output}(y)$. The corresponding y-value is the crisp output value y_{crisp} of the fuzzy controller. With U is the universe of discourse of the output variable the following equation represents the COG method:

$$y_{crisp} = \frac{\int_U (y \cdot \mu_{Output}(y))\, dy}{\int_U \mu_{Output}(y)\, dy} \tag{A.13}$$

Centre of sums:

Where the COG method determines the centre of gravity of the whole area below $\mu_{Output}(y)$, the COS method determines the centre of gravity of all separate rule results. In Figure A.7 and A.8 this means that not just one COG is calculated of the whole area, but that the COG is calculated of both areas independently, before being combined. If, in the general case, the separate rule results are the membership functions $\mu_1(y)$ ... $\mu_r(y)$ the COS method is formally given by:

$$y_{crisp} = \frac{\int_U \left(y \cdot \sum_{j=1}^{r} \mu_j(y) \right) dy}{\int_U \sum_{j=1}^{r} \mu_j(y)\, dy} \tag{A.14}$$

The difference between both methods is that overlapping areas, if such exist, are reflected more than once by the COS method.

In the two sub-sections before and this sub-section itself, all the elements of a FLC have been discussed. Often, the characteristics of a certain fuzzy controller (influenced by: fuzzification/ inference method, defuzzification method, membership function definition and rule-base) are presented in a graphical way, if is dealt with a two inputs FLC. With all the combinations of input values, is just one related output value. Plotting these combinations this leads to a 3D-graph, called the *control surface*.

A.4.5 The Takagi-Sugeno type FLC

The distinguishing mark of a *Takagi-Sugeno* FLC (TSFLC) [Takagi & Sugeno, 1985] is that fuzzy sets are used in the antecedents, but that the output consequent is, in general, a crisp linear or non-linear function and often just a constant value. The j^{th} fuzzy rule of a TSFLC is of the form

$$R_j\text{: } IF\ x_1\ is\ A_{j1}\ and\ x_2\ is\ A_{j2}\ and\ ...\ and\ is\ A_{jn}\ THEN\ y_j = f_j(x_1, x_2, ..., x_n) \tag{A.15}$$

where x_1 ... x_n are the crisp inputs, y_j is the output according to the j^{th} fuzzy rule, A_{j1} ... A_{jn} and f_j an arbitrary function.

With this type of FLC's, one can assign a control function for each region of the input space by separate rules. This makes it possible to weaken or amplify certain frequencies according to the inputs. This can also be done by conventional switching of controllers, but the fuzzy method offers a smooth transition between the individual controllers.

The final output y is the weighted combination of all the fired rules, described by

$$y = \frac{\sum_{j=1}^{r} R_j \cdot y_j}{\sum_{j=i}^{r} R_j} \quad \textbf{(A.16)}$$

where is the 'truth' value R_j of a fired rule, is determined by

$$R_j = \min\{\mu_{A_{j1}}(x_1), \mu_{A_{j2}}(x_2), \ldots, \mu_{A_{jn}}(x_n)\} \quad \textbf{(A.17)}$$

A.4.6 Benefits and limits of fuzzy control

A few of the advantages and disadvantages of fuzzy control, among others according to Driankov *et al.* [1993], are summarized in this sub-section. In Chapter 7 (Discussion and conclusions) more is said about this matter on account of the obtained simulation and real-time results and the performed comparison with conventional control methods.

Benefits:

- One main difference between fuzzy and conventional control is that a system model is not required. A FLC is developed in heuristic way with the help of expert knowledge. If a model is not available, application of fuzzy control may lead to a serious reduction of development time, since modelling can be a rather time consuming process.
- In some cases modelling might be even impossible. If any linguistic knowledge about the system is available, fuzzy control can still be applied.
- Controlling of highly non-linear systems is possible, because fuzzy controllers can be non-linear themselves.

Limits:

- As stated above, a mathematical model of the system is not necessary. However, experts knowledge and experience are. If these are not available, fuzzy control can not be applied.
- Though time is saved, since modelling is not required, an amount time may be lost during the heuristic tuning of an FLC in order to find the 'optimal' solution.
- Many applications of fuzzy control resulted in highly robust controllers, but the contrary is found as well. If fuzzy control improved robustness and stability, it is often not fully understood, and the available stability analysis methods can only be applied in special cases.