Fifth IFSA World Congress (1993), 1167-1170

CONSTRAINED DEFUZZIFICATION

Ronald R. Yager and Dimitar P. Filev

Machine Intelligence Institute, Iona College, New Rochelle, NY 10801, USA

Abstract. *We look at the problem of defuzzification in situations in which in addition to the usual fuzzy output of the controller there exists some ancillary restriction on the allowable defuzzified values. We provide two basic approaches to address this problem. In the first approach we enforce the restriction by selecting the defuzzified value through a random experiment in which the values which have nonzero probabilities are in the allowable region, this method is based on the RAGE defuzzification procedure and makes use of a nonmonotonic conjunction operator. The second approach which in the spirit of the commonly used methods, a kind of expected value, converts the problem to a constraint optimization problem.*

Introduction

Defuzzification, one of the most important issues in applications of fuzzy logic control, is associated with the selection of a crisp element based on advise contained in a fuzzy set. In previous papers related to the defuzzification problem [1, 2] we have shown that the commonly used Center of Area (COA) and Mean of Maxima (MOM) methods of defuzzification are only special cases of a more general method. This more general method is based on the BAsic Defuzzification Distribution (BADD) transformation, defining a parameterized family of defuzzified values of a given fuzzy set. In [3] we also developed a semilinear form called the Semi LInear DEfuzzification method (SLIDE). The SLIDE method of defuzzification is suitable for adaptively learning, using the algorithm of Kalman filtering, the appropriate parameter, to be used for the system being built, from the whole family of possible parameters .

In a fuzzy logic control system the output of the knowledge base is a fuzzy subset F of the universe of discourse of the output, the defuzzification process selects, based upon this set F, a crisp value y* as the controller output. Essentially the process used for the selection of y* from F is some kind of weighted averaging method [1, 2] using the membership grade of F as the basis of the weights. As the current commercial success of fuzzy logic control testifies this approach appears most reasonable.

Generally the defuzzification step is carried out in environments in which all elements of the output universe of discourse are allowable values for the crisp output y*. If no restrictions or forbidden zones in the universe of discourse are considered, the type of weighted averaging technique, such as COA or MOM, works well. In situations in which, in regards to the output space, some kind of restrictions ancillary constraints or forbidden zones exist this weighted averaging techniques brake down. In [4] Pflugar, Yen & Langari very dramatically illustrate the problem with the current averaging type defuzzification techniques.

In this paper we investigate the problem of defuzzification in the presence of restrictions on the allowed defuzzified values. We also discuss a related issue, the problem of defuzzification of nonunimodal fuzzy sets.

Defuzzification Under Constraints and Forbidden Zones

The problem of defuzzification becomes more complicated if we consider the possibility that restrictions exist in the sense that from the whole universe of discourse only some values are allowed for the defuzzified value. This case was captured by the robot example described by Pflugar, Yen & Langari [4].

Suppose we have a fuzzy logic control (FLC) knowledge base. Assume that the knowledge-base gives for a particular input an output consisting of a fuzzy subset F of the output universe of discourse X. The meaning of this fuzzy set F is that for each value $x \in X$ F(x) indicated the degree to which the rule base recommends x as the control value. As we described in the previous section this information, the fuzzy subset F, is used to to obtain a crisp control value y*.

In addition we shall consider that our problem has some restrictions on the allowable defuzzified values. The allowed defuzzified values can be expressed as another fuzzy subset H of X. In this case H(x) indicates the degree to which x is an allowed value.from the defuzzification process.

In this paper we don't discuss the nature of the allowable solutions. They can be defined from a technological point of view - e. g. certain type of control actions cannot be performed in a given setting. Allowable restrictions can be also set on the basis of control strategy - e. g. to preserve the control system from unallowable control actions. In some cases the fuzzy subset of allowable restrictions H can be a crisp set, defining a new discretization of the universe of discourse X.

The problem of defuzzification under restriction becomes one of selecting an element y* recommended by F and also allowed by H.

Random Generation Defuzzification Under Restrictions

In [2] Yager & Filev suggest an alternative to the expected value approach to the selection process. This process is a selection of y^* as the outcome generated by the performance of a random experiment guided by P. We shall call this process the **RA**ndom **GE**neration (RAGE) Defuzzification process to distinguish it from the expected valued based approach we described above. We calculate the probability distribution $P(y_i) = p_i = F(y_i) / \Sigma_j F(y_j)$ associated with output fuzzy set F. We shall assume without loss of generality that the output space consists of n elements with non-zero probability.

The RAGE defuzzification procedure is as follows:.

(1) We divide the unit interval into n intervals one for each output value y_i. We denote these intervals

$$R_i = [a_i, b_i] \qquad i = 1, \ldots, n$$

We define these R_i as

$$a_1 = 0; b_1 = p_1; \quad a_i = b_{i-1}; \quad b_i = a_i + p_i; \qquad \text{for } i > 1.$$

(2) We perform a random experiment generating a random number $r \in [0, 1]$

(3) If $r \in R_i$ then we use $y^* = y_i$.

As shown in [2] this method always results in y^* being some element for which $p_i \neq 0$, ie. $F(y_i) \neq 0$. It should also be clear that this random approach doesn't result in a averaged value for y^*.

In the following we extend this idea to the environment where there is some restriction on the allowable values for the defuzzified value. We indicate the set of allowable output values as H and use F to indicate the controller fuzzy output.

We recall that a fundamental property of the RAGE method is that the defuzzified value y^* is always one of the elements that have non-zero probability. The RAGE output is not an weighed average and as such it can easily provide a framework for implementing defuzzification in restricted environments.

A natural extension of the RAGE method appears to be one in which we combine via a conjunction the sets F and H and then use this new set to generate the probabilities to be used in the RAGE defuzzification procedure. Care must be taken in executing the requisite conjunction. If we take

$$G = H \cap F$$

we are obtaining G as a desire to get a solution that is allowable, non restricted, and recommended by the controller. Then of course

$$p_i = G(x_i) / \sum_j G(x_j)$$

Since $p_i \neq 0$ if $G(x_i) \neq 0$ we only get solutions that satisfy both.

While this simple aggregation at first seems appearing it has one dramatic drawback. Assume H and F are completely conflicting then $G = \Phi$. In this case we would get $p_i = 1/n$ for all x_i in the space. This means that we *lost* the restriction information.

Any aggregation procedure for combining F and H must keep **the priority** of the H. That is, it must not allow under any circumstances the defuzzified value to be in the forbidden region. In [5] Yager has introduced a new aggregation operator called the *nonmonotonic conjunction* which has the required priority condition.

Assume H and F are two fuzzy subsets of X. Let

$$G = \eta(H, F)$$

where

$$G(x) = H(x) \wedge (F(x) \vee (1 - \text{Poss}(F|H))).$$

We recall

$$\text{Poss}(F|H) = \text{Max}_x[F(x) \wedge H(x)].$$

We note that if F and H have at least one element in common with membership grade one, Poss(F|H) = 1, then

$$G(x) = H(x) \wedge F(x)$$

which is the usual intersection, $G = H \cap F$. At the other extreme if $F \cap H = \Phi$, then Poss[F|H] = 0 and

$$G(x) = H(x).$$

Thus when the suggestion of the controller is in complete conflict with the restriction we use the restriction. We notice in this case of complete conflict if H is a crisp subset then

$$p(x) = \frac{1}{\text{Card}(H)}$$

for all elements $x \in H$ and $p(x) = 0$ for all others.

We note that we can provide a more general formulation for the nonmonotonic intersection as

$$G(x) = H(x) \wedge S(F(x), 1 - \text{Poss}(F|H))$$

where S is any t-conorm [6, 7].

We see that the RAGE method of defuzzification under restriction consists of the following process.

Algorithm 1.

1. Aggregate the allowable region with the controller suggestion

$$G = \eta(H, F)$$

2. Normalize G to get the probabilities of the elements in the output space

$$P(x_i) = G(x_i) / \sum_j G(x_j)$$

3. Use $P(x_i)$ to randomly select an element.

Defuzzification by Constrained Optimization

The common methods used for defuzzification, COA and MOM, are based finding the <u>expected value</u> of a probability distribution over the output base set. The probability distribution used is obtained from the fuzzy output of the controller. As described by Filev & Yager [1] the difference between these two methods, COA and MOM, lies in the procedure used to obtain the probability distributions from the fuzzy sets. In [1] it is shown that with the aid of the BADD transformation we can obtain COA and MOM as special cases by simply adjusting one parameter denoted α.

With this understanding we can view these methods as the same type, finding the defuzzified value as an expected value. It should be strongly emphasized that the process of obtaining the expected value can be seen as an optimization problem.

We shall now consider the extension of the expected value based method to the problem of defuzzification under constraints. We shall assume that the fuzzy set F is the output of the fuzzy controller, it is a measure of the appropriateness of an element x as a solution, and that the fuzzy set H is the allowable region. We define

$$G = F \cap H.$$

We see G satisfies the criteria of being an good solution, F, and an allowable solution, H.

We shall now formulate the problem of defuzzification under restriction as a constrained nonlinear programming problem. As an optimization criterion we shall consider the requirement of minimization of the mean square of the error between the values of universe of discourse of G and the defuzzified value d_g. For the most generality we assume that the probabilities of the elements of the base set, the p_i's, defined by the BADD transformation [1] from the set G,

$$p_i = \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha}, \quad i=(1, n).$$

The probability distributions generated above for different $\alpha \geq 0$, are related to the membership function G. We select α to indicate the type of defuzzification we are using.

Then the criterion for a selected α has the form:

$$\text{MIN} \sum_{i=1}^{n} (x_i - d_g)^2 \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha}$$

The problem is to find d_g to minimize above equation. We note that if $\alpha = 1$ we are using a COA type method and if α is infinity then we are using a MOM type method.

We shall also include a constraint on this problem. The constraint we need is one which assures us the the defuzzified value lies in the allowable region. We shall accomplish this by requiring that $H(d_g)$ is larger then some threshold value, t_g.

Then defuzzification problem under allowable region restrictions is representable as the following constrained nonlinear programming problem:

Find the value d_g that Minimizes

$$\sum_{i=1}^{n} (x_i - d_g)^2 \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha} \qquad \text{I.}$$

Subject to

$$H(d_g) \geq t_g \qquad \text{II}$$

for a given $\alpha \geq 0$ and $t_g \in (0, 1]$. We note that α is fixed, the choice of α depends on issues discussed in [2]. The choice of t_g is fixed and it depends on how we define what it means to be contained in a fuzzy set. If H is crisp then any $t_g > 0$ works the same.

We see that the objective function can be expressed as follows

$$\sum_{i=1}^{n} (x_i - d_g)^2 \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha}$$

$$= \sum_{i=1}^{n} x_i^2 \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha} - \left(\sum_{i=1}^{n} x_i \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha}\right)^2 + \left(\sum_{i=1}^{n} x_i \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha} - d_g\right)^2 \quad \text{I'}$$

The first two terms are positive and functionally independent on d_g. The third term is also positive but depends on d_g, thus its minimum determines the minimum of the objective function. For an unconstrained d_g, the problem without condition II, the minimum coincides with d_g being the generalized defuzzified value via BADD:

$$d_g^* = \sum_{i=1}^{n} x_i \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha}$$

For the constrained defuzzification problem the minimum is obtained for those d_g, that satisfy II and minimize the third term on the right side of I', $\left(\sum_{i=1}^{n} x_i \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha} - d_g\right)^2$.

Equivalently we can use as our objective function

$$\left|\sum_{i=1}^{n} x_i \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha} - d_g\right| \qquad \text{III.}$$

Thus the solution of the constrained linear programming problem is obtained as those feasible d_g, satisfying II, that are at a **minimal** distance from the generalized BADD defuzzified value(unconstrained solution) d_g^*:

$$d_g^* = \sum_{i=1}^{n} x_i \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha}.$$

This formulation of the solution of the restricted defuzzification problem has the advantage that it doesn't require the membership function G(x) to have an analytical expression. It is enough to select only those x_i, characterized with membership grade equal or greater than the threshold t_g and to pick up the one, that is closest to the generalized BADD defuzzified value for the selected α. For $\alpha \to 0$ we receive by an arithmetic mean - like defuzzified value, for $\alpha \approx 1$ - a COA - like defuzzified value, for $\alpha \to \infty$ - a MOM – like defuzzified value.

The simplified solution of the constrained nonlinear programming problem is summarized in the following algorithm.

Algorithm 2 (*Solution for fixed α and t_g*).

1. Calculate:

$$d_g^* = \sum_{i=1}^{n} x_i \frac{g_i^\alpha}{\sum_{j=1}^{n} g_j^\alpha}$$

2. Select all x_i, i = (1, n) with membership grade $H(x_i) \geq t_g$ and form the set of indices:

$$I_h = \{i: \; H(x_i) \geq t_g\}$$

3. Find x* such that:

$$|d_g^* - x^*| = \text{Min}_{j \in I_h} |d_g^* - x_j|$$

4. Set defuzzified value to x*:

$$d_g = x^*$$

The algorithm works for fixed α and t_g. This means that the threshold has to be specified. For $t_g = 0$ defuzzified problem becomes unconstrained. The predetermined value of α defines the type of defuzzified value.

Example. Let consider fuzzy set F with membership function F(x), defined as follows:

F(x) = (0/1, .3/2, .5/3, .9/4, 1/5, 1/6, 1/7, .9/8, .8/9, .9/10, 1/11, 1/12, 1/13, 1/14, 1/15, .9/16, .8/17, .8/18, .5/19, 0/20)

and fuzzy set of allowable restrictions H with membership function H(x):

H(x) = (0/1, .2/2, .4/3, .6/4, .8/5, .8/6, 0/7, 0/8, .8/9, .7/10, 0/11, 0/12, .8/13, .8/14, .8/15, .8/16, .7/17, .6/18, .1/19, 0/20)

The intersection of H and F gives us

G(x) = (0/1, .2/2, .4/3, .6/4, .8/5, .8/6, 0/7, 0/8, .8/9, .7/10, 0/11, 0/12, .8/13, .8/14, .8/15, .8/16, .7/17, .6/18, .1/19, 0/20)

The generalized defuzzified value of fuzzy set G, without constraints on the membership grade, i.e. $t_g = 0$ is for different α as follows:

α	0	1	30
d_g^*	10.78	11.01	11.15

These values of d_g^* are not from the set of allowable values.

If $t_g = 0$ we obtain $d_g = d_g^*$. For $t_g = 0.2$, the candidates for d_g according Algorithm 1 are the following values of x_i (with membership grade $H(x_i) \geq 0.2$):

$$2, 3, 4, 5, 6, 9, 10, 13, 14, 15, 16, 17, 18$$

and consequently the set I_h is:

$$I_h = \{2, 3, 4, 5, 6, 9, 10, 13, 14, 15, 16, 17, 18\}$$

Because the distance $|d_g^* - 10|$ is minimal for $\alpha = 0$; $\alpha = 1$; $\alpha = 30$, then

$$d_g = 10$$

is the arithmetic mean - like, the COA - like and the MOM - like defuzzified value of fuzzy set G.

If $t_g = 0.8$, the candidates for d_g are the following values of x_i (with membership $H(x_i) \geq 0.8$):

$$I_H = \{5, 6, 9, 13, 14, 15, 16\}$$

Since for $\alpha = 0$ $d_g^* = 10.78$ the closest element in I_H for $\alpha = 0$ is 9, thus the *arithmetic mean-like* defuzzified value of fuzzy set G is $d_g = 9$.

For $\alpha = 1$ $d_g^* = 11.01$ the closest of the candidates for $\alpha = 1$ is 11 and since for $\alpha = 30$ $d_g^* = 11.15$ then again 11 is the closest of the candidates. Thus the *COA-like* and the *MOM-like* defuzzified values of fuzzy set G coincide at $d_g = 11$.

The result of defuzzification is depicted on Fig.

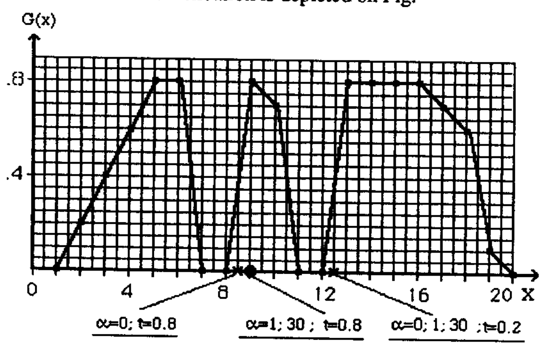

Figure 1 Defuzzification of Fuzzy set G

It is possible that in some cases two of the candidate values in I_H are equidistant from d_g^*. This situation indicates two completely equivalent. The only possibility is to randomly pick one of these alternatives. This is in fact a combination of Algorithm 1 with the defuzzification by the RAGE method. Another approach, that is out of the scope of this paper, should be the eventual extension of the rule-base to assigning preference to one of equivalent alternatives by some ancillary conditions. For example in the case of fuzzy controller priority may be given to the solution, associated with the least value of control variable as means of reducing energy.

Conclusion

A new extension of the defuzzification procedure in the presence of restrictions, important for the practical application of fuzzy systems, was discussed in this paper. Two alternative approaches to defuzzification under restriction were proposed. The first is based on the concept of defuzzification by random experiment, RAGE defuzzification. The second one leads to a constrained nonlinear programming problem. An efficient algorithm, simplifying the solution of the nonlinear programming problem was proposed.

References

[1]. Filev, D. and Yager, R. R., "A generalized defuzzification method under BAD distributions," International Journal of Intelligent Systems 6, 687-697, 1991.

[2]. Yager, R. R. and Filev, D. P., "On the issue of defuzzification and selection based on a fuzzy set," Fuzzy Sets and Systems (to appear).

[3]. Yager, R. R. and Filev, D. P., "SLIDE: A simple adaptive defuzzification method," IEEE Trans. on Fuzzy Systems, 1, 69-78.

[4]. Pfluger, N., Yen, J. and Langari, R., "A defuzzification strategy for a fuzzy logic controller employing prohibitive information in command formulation," Proceedings First IEEE Conference on Fuzzy Systems, San Diego, 1992.

[5]. Yager, R. R., "Nonmonotonic set theoretic operations," Fuzzy Sets and Systems 42, 173-190, 1991.

[6]. Alsina, C., Trillas, E. and Valverde, L., "On some logical connectives for fuzzy set theory," J. Math Anal. & Appl. 93, 15-26, 1983.

[7]. Dubois, D. and Prade, H., "A review of fuzzy sets aggregation connectives," Information Sciences 36, 85 - 121, 1985.