Which Defuzzification Is the Most Robust?

Fernando Gomide, Olga Kosheleva, and Vladik Kreinovich

Abstract When we design control systems, it is beneficial to take into account expert knowledge. An important part of expert knowledge is formulated in by using imprecise ("fuzzy") words from natural language like "small". We need to translate such expert statements into computer-understandable terms. Techniques for such translation are known as fuzzy techniques. When we use these techniques to process fuzzy inputs, we get fuzzy conclusions about the recommended control. In automatic control, we need to transform this fuzzy conclusion into a single control value that the system should apply. Such a transformation is known as defuzzification. At present, the most widely used defuzzification is so-called centroid defuzzification. It has led to many successful applications, but it has a limitation. This limitation is related to the fact that experts are not perfect, they sometimes make mistakes. With centroid defizzification, even a single expert mistake can drastically affect the resulting control. It is therefore desirable to come up with a defuzzification procedure that is as robust as possible against such mistakes. In this paper, we describe such most robust defuzzification: it turns out to be a modification of centroid defuzzification that is based on the median instead of the mean.

Fernando Gomide
University of Campinas, Campinas, Saõ Paolo, Brazil, e-mail: gomide@unicamp.br

Olga Kosheleva
Department of Teacher Education, University of Texas at El Paso, 500 W. University
El Paso, Texas 79968, USA, e-mail: olgak@utep.edu

Vladik Kreinovich
Department of Computer Science, University of Texas at El Paso, 500 W. University
El Paso, Texas 79968, USA, e-mail: vladik@utep.edu

1 Formulation of the Problem

Why fuzzy techniques: a brief reminder. In designing automated and automatic systems, it is important to take into account human expertise. Some parts of expert knowledge are clear and precise, but other parts of expert knowledge use imprecise ("fuzzy") words like "small".

Let us consider a simple example. Many people drive cars. Most of them are perfectly willing to share their driving expertise with others. From this viewpoint, it should be easy to design a self-driving car: we can just ask people how they drive, and then implement their suggestions in an automatic system. The problem is in many cases, people cannot provide precise answers to natural questions. For example, support that the car is going 100 km/h on a freeway, and the car 10 meters in front of it shows down to 95; what should a driver do? A natural answer is "brake a little bit." The problem is that while we usually understand what "a little bit" means, computers usually don't. A computer need to know for how many milliseconds and with how many kilonewtons of force does the driver hit the brake.

In the 1960s, Lotfi Zadeh, who was at that time one of the world's top specialists in automatic control, suggested that we need techniques to capture such imprecise knowledge. He called such techniques *fuzzy*, and he proposed the first version of fuzzy techniques. In this version, to describe each imprecise terms like "small" in precise computer-understandable terms, we ask the expert to assign, to each possible value x of the corresponding quantity, a value $m(x)$ from the interval $[0,1]$ that describes to what extent the value x has this property (e.g., to what extent x is small).

Some experts have trouble assigning numerical values to their degrees of belief. In such cases, an alternative *polling* way to find the value $m(x)$ is to ask several experts whether the value x satisfies the given property. If $M(x)$ out of N values say "yes", then we take $m(x) = M(x)/N$.

Whatever method we use, the resulting function $m(x)$ is known as a *membership function*, or, alternatively, a *fuzzy set*; see, e.g., [1, 2, 3, 4, 5, 7].

Need for defuzzification. Based on the fuzzy information $m_i(x_i)$ about the inputs x_i, we can determined the fuzzy information $m(y)$ about the output y – i.e., we can estimate, for each y, to what extent y is a reasonable value of the correpsonding quantity.

If our goal is to provide recommendations to a human decision maker, then this information is perfect: the decision maker will make a decision based on all these values. However, if we are designing an automatic system – such as a self-driving car – then, based on this information $m(y)$, we need to automatically select a single control value $\overline{y}$. The procedure for transforming a fuzzy set $m(y)$ into a single value $\overline{y}$ is called *defuzzification*.

Which defuzzification is mostly used now and why. To explain which defuzzifications are mostly used now, let us first consider a specific case when m experts provide m estimates $y_1, \ldots, y_m$, all with the same degree of confidence. Based on these m numbers, we need to find a single number $\overline{y}$ that base represent the given sequence of estimates: we want $\widetilde{y}_1$ to be close to $\overline{y}$, we want $\widetilde{y}_2$ to be close to $\overline{y}$, etc. We

can summarize all these closenesses by saying that we want the vector $(y_1,\ldots,y_m)$ to be close to the vector $(\overline{y},\ldots,\overline{y})$. Aa a measure of closeness between the two vectors, it is reasonable to select the usual Euclidean distance:

$$d((a_1,\ldots,a_m),(b_1,\ldots,b_m)) = \sqrt{(a_1-b_1)^2+\ldots+(a_m-b_m)^2}.$$

Since for non-negative numbers, squaring is an increasing function, minimizing the distance is equivalent to minimizing its square:

$$d^2((a_1,\ldots,a_m),(b_1,\ldots,b_m)) = (a_1-b_1)^2+\ldots+(a_m-b_m)^2.$$

In our case, this means minimizing the sum

$$(y_1-\overline{y})^2+\ldots+(y_m-\overline{y})^2.$$

According to calculus, to find the minimum of a function, we need to equate the derivative of the minimized function to zero. Differentiating the above expression and equating the derivative to 0, we conclude that

$$2\cdot(\overline{y}-y_1)+\ldots+2\cdot(\overline{y}-y_m)=0.$$

If we divide both sides of this equality by 2, and move all the terms containing y_i's to the right-hand side we conclude that

$$n\cdot\overline{y} = y_1+\ldots+y_m,$$

so

$$\overline{y} = \frac{y_1+\ldots+y_m}{m}. \tag{1}$$

In other words, the desired estimate is the mean value of the sample $y_1,\ldots,y_m$.

We can use the polling idea to extend this formula to the general fuzzy case, when we have values $y_1,\ldots,y_n$ with degrees $m(y_i) = M(y_i)/N$. This means that we have the following expert estimates:

$$(y_1,\ldots,y_1,\ldots,y_n,\ldots,y_n), \tag{2}$$

where each estimate y_i is repeated $M(x_i)$ times. So, the overall number of expert estimates is equal to $m = M(y_1)+\ldots+M(y_n)$. In the numerator of the formula (1), each term y_i is repeated $M(y_i)$ times, so the formula (1) takes the following form:

$$\overline{y} = \frac{M(y_1)\cdot y_1+\ldots+M(y_n)\cdot y_n}{M(y_1)+\ldots+M(y_n)}. \tag{3}$$

If we divide both numerator and denominator of this expression by the overall number of experts N and take into account that $M(y_i)/N = m(y_i)$, then we conclude that

$$\overline{y} = \frac{m(y_1)\cdot y_1+\ldots+m(y_n)\cdot y_n}{m(y_1)+\ldots+m(y_n)}. \tag{4}$$

One can easily see that in the above-analyzed simple case when all estimates y_i have the same degree of certainty $m(y_1) = \ldots = m(y_m)$, we can divide both numerator and denominator by this common degree and get the formula (1).

In the continuous case, when we consider all possible real values of y, the sum becomes the integral, and we conclude that

$$\overline{y} = \frac{\int m(y) \cdot y\, dy}{\int m(y)\, dy}. \tag{5}$$

The transformation (4)-(5) is known as *centroid defuzzification*. It is the most frequently used defuzzification.

Limitation of cenroid defuzzification. Fuzzy techniques using centroid defuzzification had made successful pplications, but there is a limitation to its use. This limitation comes from the fact that experts are nor perfect, they make mistakes. Some of the values y_i suggested by the experts can be mistakes. The problem is that even a single mistake can drastically change the result of centroid defuzzification. This can be illustrated already on the simple case when experts have the same degree of confidence in all the values y_i and in which, therefore, the centroid defuzzification leads to the formula (1) – i.e., to the mean value.

Suppose that we have ten experts. The actual value y is 1, nine out of ten experts provide the correct estimate $y_1 = \ldots = y_9 = 1$ except for one expert who provides an erroneous estimate $y_{10} = 100$. In this case, the mean value (1) becomes

$$\overline{y} = \frac{9 \cdot 1 + 100}{10} = 10.9,$$

which is very different from the desired value 1.

Natural question. In view of the above limitation, it is desirable to come up with defuzzificiation that should be as robust as possible against experts' mistakes.

What we do in this paper. In this paper, we describe such most robust defuzzification: it turns out to be a modification of centroid defuzzification that is based on the median instead of the mean.

2 Definitions and the Main Result

Analysis of the problem. Let us start with the above simple case when all expert estimates $y_1, \ldots, y_m$ have the same degree of confidence. In this case, the value $\overline{y}$ returned by the defuzzification procedure should depend only on these m values. Let us denote the desired defuzzification procedure by $\overline{y} = f(y_1, \ldots, y_m)$. What are natural requirements on this function?

- First, the order in which we list experts should not have any effect on the result. So, the value $\overline{y}$ should not change under any permutation.

- Second, experts may have different estimates, but they all agree that the actual value is larger than or equal to the smallest of the y_i's and that the actual value is smaller than or equal that the largest of these values; it is therefore desirable that the same two inequalities hold for the defuzzification result as well.
- Finally, robustness means that if k out of m estimates are mistakes, the result of defuzzification should still be guarantee to lie between the smallest and the largest of correct estimates.

The more expert mistakes the method can handle, the more robust it is. So, "the most robust" should mean that we have a method with the largest possible value k.

This leads to the following definitions.

Definition 1. *Let $m > 1$ be an integer. By an m-*defuzzification*, we mean a function $f(y_1, \ldots, y_n)$ of m real variables for which the following two properties hold:*

- *this function is* permutation-invariant*, i.e., for each permutation $\pi : \{1, \ldots, m\} \mapsto \{1, \ldots, m\}$, we have*

$$f(y_{\pi(1)}, \ldots, y_{\pi(m)}) = f(y_1, \ldots, y_m); \text{ and}$$

- *for all tuples $(y_1, \ldots, y_m)$, we should have*

$$\min_i y_i \le f(y_1, \ldots, y_m) \le \max_i y_i.$$

Definition 2.

- *Let $k > 0$ be an integer. We say that an m-defuzzification is k-*robust *if for all possible values of $y_1, \ldots, y_m$, we have*

$$\min(y_1, \ldots, y_{m-k}) \le f(y_1, \ldots, y_m) \le \max(y_1, \ldots, y_{m-k}).$$

- *We say that an m-defuzzification is* the most robust *if it is k-robust for some k and no m-defuzzification is ℓ-robust for any $\ell > k$.*

Discussion. Let us now describe all maximally robust defuzzifications. In this description, we will use the following standard notation for the result of sorting the values y_i in increasing order:

$$y_{(1)} \le \ldots \le y_{(i)} \le \ldots \le y_{(m)}.$$

When m is odd, i.e., when $m = 2p + 1$ for some p, the central value $y_{(p+1)}$ is known as the *sample median*.

When m is even, i.e., when $m = 2p + 2$ for some p, then there is no single central value, there are two values $y_{(p+1)}$ and $y_{(p+2)}$ which are the closest to the center of the list. In this case, in general, any value from the interval $[y_{(p+1)}, y_{(p+2)}]$ can serve as a median. We will call such value a *general median*.

Comment. In many cases, the name *median* is reserved only for the midpoint of the interval $[y_{(p+1)}, y_{(p+2)}]$.

Here is our main result.

Proposition.

- *When m is odd, i.e., when $m = 2p+1$ for some p, then the only most robust m-defuzzification procedure is the procedure that always returns the median*

$$f(y_1,\dots,y_m) = y_{(p+1)}.$$

- *When m is even, i.e., when $m = 2p+2$ for some p, then an m-defuzzification procedure is the most robust if and only if it always returns the general median, i.e., if and only if for all tuples $(y_1,\dots,y_m)$, we have*

$$f(y_1,\dots,y_m) \in [y_{(p+1)}, y_{(p+2)}].$$

Comments.

- For readers' convenience, the proof of the Proposition is placed in a special Proof section.
- In statistics, the median is also known to be the most robust estimate (see, e.g., [6]), but it should be mentioned that in statistics, a somewhat different notion of robustness is used: there, robustness means that the result remains close when the probability distribution changes.

What about the general case. In the general case, the degrees $m(y_i) = M(y_i)/N$ of different values y_i may be different. In such cases, similarly to the previous section, as the defuzzification result, we can use the result of applying the simple-case defuzzification to the sequence (2), in which each y_i is repeated $M(y_i)$ times.

In this case, the result of the most robust defuzzification is the value $\overline{y}$ for which the absolute value of the difference between the values $\sum\{M(y_i) : y_i \le \overline{y}\}$ and $\sum\{M(y_i) : y_i \ge \overline{y}\}$ is the smallest possible. Since $M(y_i) = m(y_i) \cdot N$, this is equivalent to saying that the absolute value of the difference between the values $\sum\{m(y_i) : y_i \le \overline{y}\}$ and $\sum\{m(y_i) : y_i \ge \overline{y}\}$ is the smallest possible. This means that in the continuous case, as the result of the most robust defuzzification, we select the value $\overline{y}$ for whichL

$$\int^{\overline{y}} \mu(y)\,dy = \int_{\overline{y}} \mu(y)\,dy.$$

In other words, *we select the value $\overline{y}$ that divides the area under the curve $m(y)$ into two equal parts*.

3 Proof of the Proposition

1°. It is easy to prove that every function returning a median is p-robust. So, to complete our proof, we need to prove the following two statements:

- that every p-robust function always returns a (general) median, and
- that no function is $(p+1)$-robust.

Let us prove these statements one by one.

2°. Let us first prove that every p-robust function always returns a median.

For this, we will need to prove that its result is always larger than or equal to $y_{(p+1)}$ and always smaller than or equal to $y_{(m-p)}$.

Let us prove these two statements one by one.

2.1°. Let us first prove that for every p-robust function $f(y_1,\ldots,y_m)$, we always have $y_{(p+1)} \le f(y_1,\ldots,y_m)$.

Indeed, by permutation invariance, we have $f(y_1,\ldots,y_m) = f(z_1,\ldots,z_m)$, where we selected $z_i \stackrel{\text{def}}{=} y_{(m+1-i)}$. Due to p-robustness, we conclude that

$$f(y_1,\ldots,y_m) = f(z_1,\ldots,z_m) = f(y_{(m)},y_{(m-1)}\ldots,y_{(1)}) \ge$$

$$\min(z_1,z_2,\ldots,z_{m-p}) = \min(y_{(m)},y_{(m-1)},\ldots,y_{(p+1)}).$$

By definition, the values $y_{(i)}$ are sorted in increasing order, so

$$y_{(p+1)} \le \ldots \le y_{(m-1)} \le y_{(m)}.$$

Thus,

$$\min(y_{(m)},y_{(m-1)},\ldots,y_{(p+1)}) = y_{(p+1)}$$

and so, indeed,

$$f(y_1,\ldots,y_m) \le y_{(p+1)}.$$

2.2°. Let us now prove that for every p-robust function $f(y_1,\ldots,y_m)$, we always have $f(y_1,\ldots,y_m) \le y_{m-p}$.

Indeed, by permutation invariance, we have $f(y_1,\ldots,y_m) = f(z_1,\ldots,z_m)$, where we selected $z_i \stackrel{\text{def}}{=} y_{(i}$. Due to p-robustness, we conclude that

$$f(y_1,\ldots,y_m) = f(z_1,\ldots,z_m) = f(y_{(1)},y_{(2)}\ldots,y_{(m)}) \le$$

$$\max(z_1,z_2,\ldots,z_{m-p}) = \max(y_{(1)},y_{(2)},\ldots,y_{(m-p)}).$$

By definition, the values $y_{(i)}$ are sorted in increasing order, so

$$y_{(1)} \le y_{(2)} \le \ldots \le y_{(m)}.$$

Thus,

$$\max(y_{(1)},y_{(2)},\ldots,y_{(m-p)}) = y_{(m-p)}$$

and so, indeed,

$$f(y_1,\ldots,y_m) \le y_{(m-p)}.$$

3°. Finally, let us prove that it is not possible to have a $(p+1)$-robust defuzzification.

Let us prove this by contradiction. Let us assume that such a defuzzifictaion with $k = p+1$ exists. Then, for the sequence $(y_1, \ldots, y_m) = (1, 2, \ldots, m-1, m)$, this would mean, in particular, that

$$f(1,2,\ldots,m-1,m) \le \max(y_1,\ldots,y_{n-k}) =$$

$$\max(1,2,\ldots,m-(p+1)) = m-p-1. \tag{6}$$

On the other hand, due to permutation-invariance, we have

$$f(1,2,\ldots,m-1,m) = f(m,m-1,\ldots,2,1).$$

For the tuple

$$(z_1,z_2,\ldots,z_{m-1},z_m) = (m,m-1,\ldots,2,1),$$

the $(p+1)$-robustness condition implies that

$$f(1,2,\ldots,m-1,m) = f(m,m-1,\ldots,2,1) \ge$$

$$\min(z_1,z_2,\ldots,z_{m-k}) = \min(m,\ldots,m-p) = m-p. \tag{7}$$

From (6) and (7), we conclude that

$$m-p \le f(1,\ldots,m) \le m-p-1$$

and thus, $m-p \le m-p-1$, which is not true: $m-p-1 > m-p$.

This contradiction shows that our assumption was false, and thus, indeed, it is not possible to have a $(p+1)$-robust defuzzification. The proposition is proven.

Acknowledgments

This work was supported:

- by the Brazilian National Council for Scientific and Technological Development (CNPq), Grant 304263/2024-9,
- by AT&T Fellowship in Information Technology,
- by the Institute for Risk and Reliability, Leibniz Universitaet Hannover, Germany,
- by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) Focus Program SPP 100+ 2388, Grant Nr. 501624329,
- by the European Union under the project ROBOPROX (No. CZ.02.01.01/00/22 008/0004590),

- by the Center of Excellence in Econometrics, Faculty of Economics, Chiang Mai University, Thailand,
- by the Ho Chi Minh City University of Banking, Vietnam, and
- by Thang Long University, Hanoi, Vietnam.

The authors are thankful to all the participants of the 2026 Annual Conference of the North American Fuzzy Information Processing Society NAFIPS 2026 (El Paso, Texas, March 14–16, 2026) for valuable discussions.

References

1. R. Belohlavek, J. W. Dauben, and G. J. Klir, *Fuzzy Logic and Mathematics: A Historical Perspective*, Oxford University Press, New York, 2017.
2. G. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic*, Prentice Hall, Upper Saddle River, New Jersey, 1995.
3. J. M. Mendel, *Explainable Uncertain Rule-Based Fuzzy Systems*, Springer, Cham, Switzerland, 2024.
4. H. T. Nguyen, C. L. Walker, and E. A. Walker, *A First Course in Fuzzy Logic*, Chapman and Hall/CRC, Boca Raton, Florida, 2019.
5. V. Novák, I. Perfilieva, and J. Močkoř, *Mathematical Principles of Fuzzy Logic*, Kluwer, Boston, Dordrecht, 1999.
6. D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman and Hall/CRC, Boca Raton, Florida, 2011.
7. L. A. Zadeh, "Fuzzy sets", *Information and Control*, 1965, Vol. 8, pp. 338–353.