Toward theoretical foundations of type-n fuzzy techniques: why they are effective, when they are effective, and what is the most efficient way to use them

Olga Kosheleva, Vladik Kreinovich, Patricia Melin, and Oscar Castillo

Abstract In the original fuzzy technique, the expert describes his/her degree of confidence in their imprecise ("fuzzy") statement by a number from the interval $[0,1]$ – so that 1 means absolute confidence, 0 means absolute lack of confidence, and intermediate values describe intermediate degrees of confidence. This technique has many useful practical application, but it has a serious limitation: just like an expert is unable to describe the exact value of the corresponding physical quantity, the same expert is unable to describe his/her degree of confidence by a exact number. A natural solution to this challenge is to allow the expert to describe his.her degree not by a number, but by a natural-language words like "very confident" or "somewhat confident". This approach is known as type-2 approach. It provides a more adequate description of expert knowledge, but it requires more computations to handle. As a result, sometimes it leads to better application results, and sometime, more traditional techniques (called type1) work better. This raises several natural questions: why they type-2 techniques effective, when they these techniques effective, and what is the most efficient way to use these techniques. In this paper, we overview theoretical results that provide (at least partial) answers to these questions.

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

Patricia Melin and Oscar Castillo
Division of Graduate Studies and Research, Tijuana Institute of Technology
Calzada del Tecnologico 12950 Tomas Aquino, CP 22414 Tijuana, B.C., México,
e-mail: {ocastillo,pmelin}@tectijuana.mx

1 Formulation of the problem

Need for expert knowledge, need for fuzzy techniques. A significant part of our knowledge comes from experts: expert teachers teach students, expert doctors treat patients, expert pilots fly planes, expert engineers design spaceships, etc.

It is not possible to have the best doctor treat all the patients – there are millions of patients. It is not possible to have the best pilot fly al the planes – there are thousands of planes. It is therefore desirable to incorporate the knowledge of the best experts into a computer-based system that would help other, less skilled experts – and maybe even help population as a whole. Some of this knowledge can be described in precise terms, terms that are easy to incorporate into a computer-based system.

However, a significant part of this knowledge is formulated by experts via using imprecise ("fuzzy") natural-language words like "small". We humans more or less understand such words, but for computers, understanding such words is a big challenge: their natural language is numbers, not words. So, we need techniques that would translate such knowledge into precise computer-understandable terms. The need for such translation techniques was first well understood by Lotfi Zadeh from the University of California – Berkeley, who was at that time one of the world's leading specialists in automated control. He called such translation techniques "fuzzy", and he proposed the first version of such fuzzy techniques, the version that has since been developed into a successful application technique; see, e.g., [3, 17, 29, 32, 33, 39].

Main idea behind the original fuzzy techniques. Zadeh's main idea was best described by Zadeh himself: he liked to repeat that "everything is a matter of degree". From this viewpoint, to describe what the expert means by, e.g., "small", we need to ask the expert to describe, for each possible value x of the corresponding physical quantity, the degree $m(x)$ to which, in the opinion of this expert, this value x is small. Zadeh called the resulting function $m(x)$ *membership function* or, alternatively, a *fuzzy set*.

In a computer, "true" is usually described by 1, and "false" by 0. So, it is reasonable to describe intermediate degrees of confidence by numbers between 0 and 1. This is exactly what Zadeh proposed, and this is exactly what led to many empirical successes.

Limitations of the original fuzzy techniques. Once we elicit the degrees from the expert, fuzzy techniques provide a smooth sailing. The main problem is eliciting these degrees.

It is a problem since, similarly to how experts often cannot formulate their knowledge in precise terms, the experts also cannot describe their degree of confidence by a precise number. For example, we can usually distinguish between confidence 90% and 80% – but no one can seriously distinguish between confidence 80% and confidence 81%.

How to overcome this challenge: enter type-2 (and, more generally, type n) techniques. It is difficult for experts to describe their degree of confidence that a

value x satisfies the gievn imprecise property (like "small") by an exact number $m(x)$. It is more natural for them to describe it by a natural-language term such as "very confident", "somewhat confident", etc. Then, we can use the usual fuzzy techniques to transform this term into a fuzzy set, i.e., into a membership function $m(x,d)$ that assigns, to each value x of the corresponding physical property and to each value $d \in [0,1]$, the degree to which the value d adequately describes the expert's confidence that x satisfies the given property.

In this arrangement – originally proposed by Zadeh himself – we have a membership function $m(x)$ whose value, for each x, is also a membership function, Thus, this approach is known as *type-2* fuzzy approach. This approach had many successful applications; see, e.g., [29].

We can go even further and allow fuzzy degrees $m(x,d)$ instead of exact numbers – this will make it type-3 technique. This technique has also been successfully used; see, e.g., [5, 6, 7, 8, 22].

Remaining challenge. Type-2 (and, more generally, type-n) techniques provide a more adequate description of expert knowledge, but they requires more computations to handle. As a result, sometimes the use of type-n techniques leads to better application results, and sometimes, more traditional techniques (called type-1) work better.

This raises several natural questions: why are type-2 techniques effective, when are these techniques effective, and what is the most efficient way to use these techniques.

What we do in this chapter. In this chapter, we overview theoretical results that provide (at least partial) answers to these questions.

Of course, we do not promise to solve all the related problems, what we present are general theoretical foundations that will hopefully eventually lead to final answers.

Comment. Due to size limitations, in this chapter, we do not provide all the technical details of the corresponding algorithms and proofs. We describe the main ideas, and the details can be found in the papers that we cite.

Structure of the chapter. In the following three sections (Sections 2–4), we provide possible answers to the above three questions. In the remaining chapters, we deal with other foundational issues related to type-n fuzzy uncertainty:

- approached in between type-1 and type-2 in Section 5,
- uses of type-2 techniques beyond data processing in Section 6, and
- uses of type-2 ideas beyond fuzzy uncertainty in Section 7.

2 Why are type-2 (and, more generally, type-n) techniques effective?

How expert knowledge is usually formed: localized structure of expert information. To start explaining why type-n techniques are often effective, let us first recall how expert knowledge is usually formed. Experts have experience in dealing with many individual cases. Usually, when an expert had several similar cases and in all (or at least in most of) these cases, a similar solution worked, the expert formulates – and recommends – a general rule for all future similar cases. (Sometimes, this is generalized further, when it turns out that the same recommendation can be used for several different groups of situations.)

A situation is usually described by the values of several quantities $x_1, \ldots, x_n$. In these terms, two situations $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$ are similar if the corresponding values are close: $x_i \approx y_i$ for all i. Crudely speaking, we divide the range of each of the quantities into several (L) subranges. Each typical situation is then described by specifying one of the L subranges for each of n quantities. The number of such typical situations is thus equal to L^n.

Resulting types of possible explanations. Because of the localized form of expert knowledge, there are two possible ways to explain the success of type-n techniques:

- we can have a more general type-n explanation – an explanation that uses only a few specifics of type-n techniques, and largely focuses on localized character of type-n approaches; and
- we can also have more specific explanations, that actively use several specifics of fuzzy type-n techniques.

In this section, we will give examples of both types of explanations.

What specifics can be used in more general explanations. Mostly, more general explanations use the fact that in describing type-1 fuzzy rules, we usually only use simple membership functions – such as triangular or trapezoid. This use is justified by empirical success: in many practical situations, the use of these membership functions leads to the best results. This use can also be theoretically justified – one of such justifications is given in Section 6.

This means that for each typical situation, we have only a few parameters of the resulting fuzzy input. Let us denote the number of such parameters by C.

The first of more general explanations. One of the main advantages of fuzzy techniques is that they are related to natural-language descriptions and are, thus, explainable. This is a big advantage, especially taking into account that lack of explainability is one of the major problmes of the modern neural network approaches; see, e.g., [14]. But, of course, fuzzy-related explainability has limits:

- if we use a few rules and arguments in making a decision, then a user can easily understand where this decision came from;

- on the other hand, if we use thousands and millions of natural-language rules to make a decision, then, while each rule may be understandable, the whole conclusion is not explainable at all.

One – rather serious – limit on such explainability comes from the well-known "seven plus mimus two" rule (see, e.g., [30, 35]), according to which, in particular, a person can meaningfully divide a region into no more than 7 plus minus 2 subregions:

- some people divide into $7-2=5$ subregions,
- some people divide into $7+2=9$ subregions.

To make the subdivisions understandable to everyone, we need to use no more than 5 subdivisions: $L \le 5$. Thus, the largest number of typical situations that we can cover this way is 5^n. For each typical situation, a type-1 approach specifies C parameters. Thus, overall, if we want an explainable description with the type-1 approach, we can use no more than $C \cdot 5^n$ parameters.

In general, we want to describe all possible types of practical cases, in which we can have all kinds of functions $f(x_1,\ldots,x_n)$ – functions that describe the action appropriate for a given situations. To exactly describe a general function, we need to have infinitely many parameters: for example, the values of this function at infinitely many possible tuples $x=(x_1,\ldots,x_n)$. If we limit the number of parameters, then we can only describe the desired function with some accuracy – and the more parameters we use, the closer we are to the desired function.

So, if the approximation accuracy that we achieve by using $C \cdot 5^n$ parameters is not sufficient, the only way to have a more accurate description (and still keep the description explainable) is to use more than C parameters to describe the recommendation for each typical situation. This means going beyond the usual type-1 approach in which we can only use C parameters – so in this case, type-2 (and, more generally, type-n) approach is more appropriate.

The second of more general explanations. The above explanation shows that sometimes, we cannot reach the desired accuracy with the type-1 approach, we have to use the type-2 approach. As we will show in this subsection, this does not necessarily mean that when it is possible to achieve the desired accuracy with the type-1 approach, we should use it. We will show that in some such cases, type-2 (and type-n) techniques are still sometimes better.

Comment. The main ideas of this explanation first appeared in [20].

To describe how much accuracy we can reach by using a given number of parameters, let us use the experience of physics [13, 38]. In physics, we often need to have a good approximation to an unknown function, and the usual way to do it is to expand the unknown function into Taylor series and use only the first few terms in this expansion. Empirically, this approach works the best, so let us use this approach here as well – to describe how we approximate the desired function in each combination of subranges. Let us assume that we can afford the total of V variables. We can arrange them differently:

- we can have a few subdivisions, with small L, and have more parameters V/L^n to describe the function in each of L^n combination of subranges, and
- we can have many subdivisions, with larger L, which leaves us with fewer parameters V/L^n to describe the function on each typical situation.

In both cases, there is a certain order P of a polynomial approximation that can be achieves by using these parameters – and thus, there is the corresponding accuracy. Let us find out which arrangement provides the most accurate description.

To get a full order P polynomial of n variables, we need to consider the coefficients at all possible terms $x_1^{d_1} \cdot \ldots \cdot x_n^{d_n}$ for which $d_1 + \ldots + d_n \le P$. To count the number of all such terms, let us take into account that we can put such terms in 1-1 correspondence with the lists in which we have:

- d_1 zeros,
- then a separating one,
- then d_2 zeros,
- then a separating one,
- $\ldots$,
- then d_n zeros,
- then a separating one,
- and, finally, $P - (d_1 + \ldots + d_n)$ zeros.

Each such sequence has P zeros and n ones, to the total length of $P+n$. Thus, the overall number of such terms is equal to the overall number of possibilities to select P out of $P+n$ elements, i.e., to:

$$\binom{P+n}{n} = \frac{(P+n)!}{P! \cdot n!}.$$

This is the number of parameters that we need to describe each typical situation. So, to describe all L^n possible typical situations, we need to use

$$V = L^n \cdot \frac{(P+n)!}{P! \cdot n!}$$

parameters.

What is the accuracy of this approximation? When we limit ourselves to polynomials of order P, this means that we ignore all the terms of orders $P+1$ or higher. Thus, on each subrange, the relative approximation error can be described by a usual term of order $P+1$, i.e., a term $(x_1 - x_{01})^{d_1} \cdot \ldots \cdot (x_n - x_{0n})^{d_n}$, where x_{0i} is the center of the i-th subrange, and $d_1 + \ldots + d_n = P+1$.

When we divide the range of each variable x_i into L subranges, each subrange has width $1/L$. So the largest possible value of each difference $x_i - x_{0i}$ is $1/(2L)$, and the largest possible value of the approximation error is

$$\Delta = \frac{1}{(2L)^{P+1}} = \frac{1}{2^{P+1}} \cdot \frac{1}{L^{P+1}}.$$

For a given overall number of parameters V and a given degree P, we can determine $1/L$ from the equation containing V as

$$\frac{1}{L} = \frac{1}{V^{1/n}} \cdot \left(\frac{(P+n)!}{P! \cdot n!} \right)^{1/n}.$$

Substituting this expression into the formula for Δ, we conclude that

$$\Delta = \frac{1}{2^{P+1}} \cdot \frac{1}{V^{(P+1)/n}} \cdot \left(\frac{(P+n)!}{P! \cdot n!} \right)^{(P+1)/n}.$$

The optimal degree P is the one for the approximation eroor Δ is the smallest.

A simple analysis (see, e.g., [20]) shows that when V increases, the optimal value P also increases – which means that for each typical situation, it is better to have more parameters to describe the recommendation for this situation. For sufficiently large V, this number is larger than what the type-1 approach provides – so we need to use type-2 and type-2 approaches.

A more specific explanation. A more specific explanation is provided in a recent paper [36]. This explanation is based on the fact that the more models we have, the more accurately these models can represent different functions.

This natural fact can illustrated on the simplest example of approximating numbers from the interval $[0,1]$.

- If we can only have one model, then the best idea is to take $x_1 = 1/2$, then every number from this interval can be approximated with accuracy 1/2.
- If we can have N models, then the best idea is to divide the interval $[0,1]$ into N equal subintervals $[0,1/N]$, $[1/N,2/N]$,..., and take as the points, central points of these subintervals, This way, we can approximate any number with the accuracy $1/(2N)$.

In general, the more models we have, the more accurate the approximation.

Let us apply this idea to the comparison between type-1 and type-2 models. In the corresponding explanation, we allow membership functions that are more general than triangular ones. To describe how a membership function behaves on each an interval $[\underline{x},\overline{x}]$ where this function is increasing, we need to describes its values $m_1,\ldots,m_n$ on the uniformly distributed points $x_1 = \underline{x} < x_2 < \ldots < x_n = \overline{x}$ from this interval. This way, we need n parameters. However, not all combinations of these parameters are possible: since the function is increasing, these values need to satisfy the inequalities $m_1 \le m_2 \le \ldots \le m_n$.

This is what we need if we use the type-1 approach. What if we use the type-2 approach, namely, the simplest case of this approach when we use interval-valued fuzzy. In this case, for each point x, the expert's degree of confidence is described by an interval $[\underline{m}(x),\overline{m}(x)]$ – the interval is the simplest case of the fuzzy set, when all confidence degrees are 0s or 1s.

In this approach, we need two parameters to describe each value $m(x)$. So, if we have n parameters, we can only describe the values $\underline{m}_i$ and $\overline{m}_i$ at $n/2$ points. These values must satisfy the following inequalities:

- the inequality $\underline{m}_1 \le \underline{m}_2 \le \ldots$ about the lower bounds,
- the inequality $\overline{m}_1 \le \overline{m}_2 \le \ldots$ about the upper bounds, and
- the inequalities $\underline{m}_1 \le \overline{m}_1, \underline{m}_2 \le \overline{m}_2, \ldots$, describing the relation between the lower and upper bounds.

It can be shown that these inequalities provide fewer restrictions on n numbers that the type-1 restrictions. Indeed:

- in addition to the case when we have

$$\underline{m}_1 \le \overline{m}_1 \le \underline{m}_2 \le \overline{m}_2, \ldots$$

  the case that covers all type-1 sequences,
- we can also have many other sequences for which, e.g., $\overline{m}_1 > \underline{m}_2$.

So, even when the number of parameters is the same, in the type-2 case, we have more combinations of these parameters – thus, more possible models and therefore, more accurate approximation of the desired function.

3 When are type-2 (and, more generally, type-n) techniques effective: a brief discussion

General answer. As we have mentioned, in some cases, type-2 (and type-n) techniques are better, while in other cases, type-1 methods work better. How can we decide which approach to use? Some criteria were given in the previous section:

- if the approximation error of the explainable type-1 model is too high for our purposes, then we clearly need to use type-2 or type-n approaches;
- if the optimal number of parameters-per-typical-situation (as computed by the formula from the second of more general explanations), is larger than the number of type-1 parameters, then we also need to use type-2 or type-n models.

Comments. For the 1-D case, recommendations on when to use type-1 or type-2 are provided in [20].

Remaining open question – and partial answer to this question. In the previous section, we compared situations in which we:

- either have either different number of parameters per typical situation,
- or different constraints of the corresponding parameters.

In some cases, however, we have two techniques in which we have the same number of parameters and the same constraints on the parameters. In many such situations, deciding which of these two techniques is better is still a theoretical challenge.

A partial answer to this challenge is provided in [24], where we compared the interval-valued fuzzy techniques with type-2 fuzzy techniques that use Gaussian membership functions. In both cases, we need two parameters to describe the corresponding membership function $m(x)$:

- in the interval case, we need to describe the midpoint $\widetilde{m}$ and the half-width Δ of the interval $[\underline{m},\overline{m}]$:
$$\widetilde{m} = \frac{\underline{m}+\overline{m}}{2},\ \ \Delta = \frac{\overline{m}-\underline{m}}{2};$$
- in the Gaussian case, we need to describe the parameters m_0 and σ of the corresponding membership function
$$\exp\left(-\frac{(m-m_0)^2}{2\sigma^2}\right).$$

In both cases, we need two parameters:

- we need $\widetilde{m}$ and Δ in the interval case, and
- we need m_0 and σ in the Gaussian case.

In both cases, the only constraint is that the second parameter should be non-negative. In this specific case, [24] explains when each of the two approaches leads to better results.

4 What is the most efficient way to use type-2 (and, more generally, type-n) techniques?

Why do we need fuzzy data in the first place. To understand how to efficiently use type-2 fuzzy data, let us recall where the need for data processing comes from. In general, we process data to make an appropriate decision. We consider situations in which know how to make this decision in situations when we know the values $x_1,\ldots,x_n$) of some physical quantities. In some cases, we can measure these quantities and get their values. However, sometimes, such a measurement is not possible – or at least too complicated.

In such cases, we have to rely on expert estimates – which lead to a fuzzy description of the inputs x_i.

Resulting problem: a general formulation. So, we face the following situation: we have an algorithm $y = f(x_1,\ldots,x_n)$ that transforms real values x_i into the desired value y. If we have only fuzzy information about the inputs x_i, what can we conclude about y? The corresponding problem is called *data processing under fuzzy uncertainty*.

How do we process fuzzy data in type-1 case? To understand how to best process type-2 data, let us recall how data processing is performed in the traditional type-1

fuzzy case. For that purpose, we need to recall the notions of "and"-operations (also known as t-norms) and "or"-operations (also known as t-conorms).

As we have mentioned, one of the main ideas of fuzzy techniques is that we elicit, from the expert, the expert's degrees of confidence that each possible value x of a physical quantity satisfies the corresponding imprecise property – e.g., that this value is small. The problem is that in actual decisions, we usually do not just take into account the value of a *single* variable.

For example, when we drive, we take into account not only our own speed and location, but also speeds and locations of other cars and other objects – and distances from our car to all these other objects. We may have a rule like

"if a car closely in front slows down a little bit, then break a little bit".

We can elicit, from the expert driver, what he/she means by "closely" about distances and what he/she means by "slows down a little bit", what we really need is the degree to which both closeness and slowing down are true.

From the purely theoretical viewpoint, we can also ask the expert about all possible combination of distance and change-in-speed, but there are too many such combinations to make it practical. And if we need to take into account five or six inputs, the number of possible combinations becomes unrealistically astronomical.

Since we cannot, in general, elicit the degree of confidence in such a complex statement of the type $A \,\&\, B$, we therefore need to estimate this degree based on what we have already elicited, i.e., based on the degrees a and b of the statements A and B. The algorithm $f(a,b)$ that transforms these values a and b into the desired estimate is called an *"and"-operation* or, alternatively, a *t-norm*. The most widely used t-norms are $f_{\&}(a,b) = \min(a,b)$ and $f_{\&}(a,b) = a \cdot b$.

Similarly, we need an algorithm that estimates our degree of confidence in an "or"-statement $A \vee B$. Such an algorithm is known as an *"or"-operation* or a *t-conorm.* The most widely used t-conorms are $f_{\&}(a,b) = \max(a,b)$ and $f_{\&}(a,b) = a + b - a \cdot b$.

Let us show how this can be used in data processing under fuzzy uncertainty. Indeed, when is number Y a possible result of data processing? When there are some $X_1, \ldots, X_n$ for which:

$$X_1 \text{ is possible, } \textit{and } X_2 \text{ is possible, } \textit{and } \ldots, \text{ and } Y = f(X_1, \ldots, X_n).$$

The degree to which X_i is possible is described by the corresponding membership function as $m_i(X_i)$. If we use the simplest "and"-operation minimum, then for tuples $X = (X_1, \ldots, X_n)$ for which $Y = f(X_1, \ldots, X_n)$, the degree of confidence in a centered statement is equal to $\min(m_1(X_1), \ldots, m_n(X_n))$.

The value Y is possible if:

- either the above centered statement holds for one of such tuples,
- *or* this statement holds for another tuple, etc.

There are infinitely many such tuples, so we need to use an appropriate "or"-operation to combine infinitely many degrees corresponding to different tuples. Most known "or"-operations will lead us to the meaningless answer 1 when we

apply them infinitely many times. The only known "or"-operation that does not lead 1 is the maximum. So, we conclude that the degree of confidence $m(Y)$ that Y is the value of the data processing is equal to

$$m(Y) = \max\{\min(m_1(X_1), \ldots, m_n(X_n)) : f(X_1, \ldots, X_n) = Y\}.$$

This formula was derived first by Zadeh himself, it is known as *Zadeh's extension principle*.

From the computational viewpoint, it looks complex, but the corresponding computations can be drastically simplified if we represent each membership function $m_i(x_i)$ by its so-called α*-cuts* $\mathbf{m}_i(\alpha) \stackrel{\text{def}}{=} \{x_i : m_i(x_i) \geq \alpha\}$.

When is $m(Y) \geq \alpha$? The largest of several values is larger than or equal to α if and only if one of them is greater than or equal to α, i.e., if and only if $\min(m_1(X_1), \ldots, m_n(X_n)) \geq \alpha$ for some tuple X.

When is the minimum of several numbers larger than or equal to α? When each of them is $\geq \alpha$. So, $m(Y) \geq \alpha$ if and only if we have some values X_i for all of which $m_i(X_i) \geq \alpha$ and $Y = f(X_1, \ldots, X_n)$. In other words, for every α:

$$\mathbf{m}(\alpha) = f(\mathbf{m}_1(\alpha), \ldots, \mathbf{m}_n(\alpha)) \stackrel{\text{def}}{=} \{f(X_1, \ldots, X_n) : X_i \in \mathbf{m}_i(\alpha)\}.$$

So, the α-cut of y is the range of possible values of $f(x_1, \ldots, x_n)$ when each x_i is in the corresponding α-cut.

The α-cuts are usually intervals. For intervals, the computation of the corresponding range is known as *interval computations*. There exist many efficient interval computation tools; see, e.g., see, e.g., [15, 26, 28, 31].

How to extend this to type-2 and type-n cases. How can we extend this to the type-2 case? In the type-2 case, the values $m_i(x_i)$ are themselves membership functions. So, to compute the α-cut $\mathbf{m}(y, \alpha) = [\underline{m}(y, \alpha), \overline{m}(y, \alpha)]$ of the fuzzy value $m(y)$, we need to find the range of the right-hand side (of the above formula that describes $m(y)$ in terms of $m_i(x_i)$) when each value $m_i(x_i)$ belongs to the corresponding α-cut $\mathbf{m}_i(x_i, \alpha) = [\underline{m}_i(x_i, \alpha), \overline{m}_i(x_i, \alpha)]$. The above expression of $m(y)$ in terms of the values $m_i(x_i)$ is monotonic in $m_i(x_i)$, so:

- its largest value is attained when the values $m_i(x_i)$ are the largest possible, and
- its smallest value is attained when the values $m_i(x_i)$ are the smallest possible.

Thus,

$$\overline{m}(Y, \alpha) = \max\{\min(\overline{m}_1(X_1, \alpha), \ldots, \overline{m}_n(X_n, \alpha)) : f(X_1, \ldots, X_n) = Y\}$$

and

$$\underline{m}(Y, \alpha) = \max\{\min(\underline{m}_1(X_1, \alpha), \ldots, \underline{m}_n(X_n, \alpha)) : f(X_1, \ldots, X_n) = Y\}.$$

So, for each α, we arrived at, in effect, the same formulas as in the type-1 case – and we already know how to use interval computations to perform these computations.

The only difference that if we use, as usual, 11 possible values of α: 0, 0.1, 0.2, ..., 0.9, and 1, then:

- in the type-1 case, we need to repeat interval computations 11 times, while
- in the type-2 case, we need to have 11 repetitions for each of 11 possible values α, i.e., overall we need 11^2 interval computations.

Similarly, we can deal with the case when we have type-3 – in which case we need 11^3 interval computations, type-4, etc.

Detailed general formulas are given in [19, 25], and even more detailed formulas are given in [22] specifically for the type-2 and type-3 cases.

5 In between type-1 and type-2 fuzzy techniques

Why do we need to have something in between type-2 and type-2? In the previous sections, we analyzed issues related to the use of type-n fuzzy techniques to process data. In particular, we dealt with the fact that, while type-n techniques provide a more accurate description of expert knowledge, their use requires more computations.

What if we, due to the increased computational complexity, cannot afford to use the whole type-2 techniques – or even its interval-valued version – but we still want to use some advantages of the type-2 approach? Let us see what we can do.

Analysis of the problem. In the traditional type-1 approach, for each value x of the estimated quantity, we ask the expert to provide the degree m to which this value is possible. For example, if the expert believes that the temperature on a certain day was hot, we ask the expert to provide, e.g., the degree m to which 27 C is hot, in the opinion of this expert. In the type-2 approach:

- we allow the expert to provide several different degrees $m_1, \ldots, m_n$, and
- for each of these degrees m_i, we ask the expert to provide the degree d_i to which the degree m_i sounds reasonable.

If we cannot afford using different values m_i, if we can only afford a single value m, then, to use the general idea of type-2, it makes sense to also ask the expert to provide the degree d to which this expert is confident in his/her estimate m.

This leads us exactly to Z-numbers. Interestingly, we arrive at yet another idea proposed by Zadeh himself: in addition to the fuzzy knowledge, the expert should provide a degree to which he/she is confident about this knowledge. For example, a witness may say that a person he saw at the scene of a robbery was tall, but this witness may not be 100% sure about it, he/she may say that he/she is 80% confident. Such pairs of fuzzy knowledge and the corresponding degree of confidence are called *Z-number*, after Zadeh; see, e.g., [1, 2, 27, 40].

What is the relation between Z-numbers and type-2 fuzzy sets? Based on our description of how we get Z-numbers from type-2, to get a full type-2, we need to

provide several Z-numbers. This natural conclusion can be mathematically justified: in [2], it is proven that under a reasonable condition of monotonicity, every type-2 fuzzy set can be represented as a result of applying an appropriate data processing algorithm to some Z-numbers.

6 Type-2 fuzzy beyond data processing

Fuzzy techniques are more general that their use in data processing. In the previous sections, we mostly talked about using fuzzy techniques in data processing. However fuzzy techniques are more general than that. Indeed, the main idea of fuzzy techniques is that they transform imprecise ("fuzzy") natural-language expert statements into precise computer-understandable form. In the previous sections, we considered applications of fuzzy techniques to data processing, when the expert knowledge is about the inputs: e.g., we do not know the actual temperature, but we have an expert statement that it is somewhat hot.

But expert statements can be not only about data, they can also be about techniques that can be used to process data. In this case, fuzzy techniques can help transform imprecise heuristic ideas into precise algorithms. Traditional (type-1) fuzzy techniques have indeed been successfully used for this purpose; see, e.g., [4, 9, 10, 11].

Since type-2 techniques provide a more adequate description of expert knowledge, a natural idea is to use type-2 techniques for this purpose. And type-2 techniques have indeed been successfully used; see, e.g., [18]. Interesting, this was an application to select appropriate fuzzy techniques: namely, it explains the ubiquity of triangular and trapezoid membership functions. Following [18], let us briefly explain this application.

Type-2 fuzzy analysis explains ubiquity of triangular and trapezoid membership functions. In principle, different shapes of membership functions are possible. However, empirically, the most widely use are triangular and trapezoid members – because empiricaly, their use leads, in most cases, to better results. How can we explain the ubiquity of such membership functions?

We want a membership function $m(x)$ that describes to what extent the value satisfies some fuzzy property – e.g., to what extend x is large, or to what extent the temperature x is hot. Usually, we know the value $\underline{x}$ that definitely does not satisfy this propoerty – i.e., for which $m(\underline{x}) = 0$. We also usually know the value $\overline{x}$ that definitely satisfies this property – i.e., for which $m(\overline{x}) = 1$. The question is which degrees $m(x)$ are reasonable for values x between these two values, i.e., for values x from the interval $[\underline{x}, \overline{x}]$.

The answer to this question (the answer described, in detail, in [18]) is based on the following natural statement about the membership function $m(x)$:

"if x and x' are close, then $m(x)$ and $m(x')$ should be close".

This is a fuzzy statement about membership degrees. In this sense, it is a type-2 statement – although, of course, it is different from type-2 fuzzy sets that are used in data processing.

The closeness between the two values a and b is characterized by the distance $|a-b|$ between them. So, the degree to which a and b are close can be described as $d(|a-b|)$, for some decreasing function $d(x)$.

From the purely mathematical viewpoint, there are infinitely many real numbers between $\underline{x}$ and $\overline{x}$. However, in practice, we only measure x with some accuracy h, so values whose difference is smaller than h are not distinguishable from each other. From this point, it is sufficient to only consider distinguishable points x, i.e., points

$$x_0 = \underline{x}, x_1 = \underline{x} + h, x_2 = \underline{x} + 2h, \ldots, x_i = \underline{x} + i \cdot h, \ldots, x_n = \underline{x} + n \cdot h = \overline{x},$$

for an appropriate n. In this description, the above statement takes the following form:

- the values $m(x_0)$ and $m(x_1)$ are close, and
- the values $m(x_1)$ and $m(x_2)$ are close, and
- $\ldots$, and
- the values $m(x_i)$ and $m(x_{i+1})$ are close, and
- $\ldots$, and
- the values $x(x_{n-1})$ and $m(x_n)$ are close.

If we use min as the "and"-operation, then the degree D to which the values $m(x_i)$ satisfy this property is equal to

$$D = \min(d(|m(x_0) - m(x_1)|), \ldots, d(|m(x_i) - m(x_{i+1})|), \ldots, d(|m(x_{n-1}) - m(x_n)|)).$$

It is reasonable to select the membership values that maximally satisfy this property, i.e., for which this degree D is the largest. It turns out that such values come from a linear function $m(x)$. This explains why piece-wise linear functions – like triangular or trapezoid ones – are indeed ubiquitous.

Interestingly, the result remains the same if instead of minimum, we use another widely use "and"-operation – algebraic product $a \cdot b$.

7 Type-2 beyond fuzzy

Why do we need to go beyond fuzzy. Many quantities are known with uncertainty. Fuzzy uncertainty is only one of the possible types of uncertainty.

Indeed, our knowledge about the world comes from two main sources: measurements and expert estimates. A large part of expert knowledge comes in the form of imprecise ("fuzzy") natural-language statements, statements that are difficult for computers to understand. Fuzzy techniques are, by definition, techniques that transform such knowledge into computer-understandable terms.

But a lot of knowledge also comes from measurements, from a process that generates not natural-language words but numbers. The fact that the results of measurements are numbers does not mean that there is no uncertainty. The uncertainty is that the measurements are never 100% accurate (see, e.g., [34]): the value $\widetilde{x}$ produced by a measurement is, in general, different from the actual (unknown) value x of the measured quantity. In other words, the difference $\Delta x \stackrel{\text{def}}{=} \widetilde{x} - x$, known as *measurement error*, is, in general, different from 0.

Case of probabilistic uncertainty. As anyone who ever measured anything knows, not only the measurement result differs from the actual value, but of we repeat the measurement several times, we get slightly different results. We cannot predict the value of the measurement error, but by observing it many times, we can find the frequency with which different values Δx appear. In mathematical terms, what we find is the *probability* distribution on the set of measurement errors. As a result, once we get the measurement result $\widetilde{x}$, we do not get the actual value x of the measured quantity, but we get a probability distribution on the set of possible values x.

This situation is known as *probabilistic uncertainty*. This probabilistic approach is what our students learn when they learn how to measure and how to process measurement results.

Limitations of the traditional probabilistic approach to uncertainty. The traditional probabilistic approach implicitly assumes that we can exactly reconstruct the probability distribution of the measurement error from finitely many measurements. This assumption is, of course, an idealization. To exactly describe a probability distribution, we need to know infinitely many parameters – e.g., the values of the probability density function $f(x)$ for all infinitely many possible values x. We cannot determine infinitely many parameters cased on the results of finitely many experiments. So what can we do?

Interval-values and type-2 probabilities. Since we cannot uniquely determine the probability distribution, we thus have a whole set of possible probability distributions. This is a particular case of what is known as *imprecise probability* approach. In this case, for each event, different possible probability distributions lead, in general, to different values for the probability of this event. Usually, these values form an interval, so we get what is usually known as *interval-valued* probability approach.

For example, instead of the exact values of the cumulative distribution function $F(a) \stackrel{\text{def}}{=} \text{Prob}(x \le a)$, we get an interval of possible values $[\underline{F}(a), \overline{F}(a)]$. Such interval-valued version of the cumulative distribution function is known as the *probability box*, or, for short, *p-box*; see, e.g., [12]. Efficient algorithms have been developed for processing p-box uncertainty.

In this approach, we only take into account which probability distributions are possible and which are not. But we can go further. We can consider many such situations and determined the frequency with which different probability distributions appear. In other words, we can consider probability distribution of the set of all possible probability distributions. This is a direct probabilistic analog of type-2 fuzzy, so it is reasonable to call such approach *type-2 probabilities*; see, e.g., [21]. This

idea is (somewhat implicitly) used in the so-called *Bayesian* approach to statistics (see, e.g., [16]).

Type-2 probabilistic approach is different from the usual (type-1) approach. At first glance, it may seem that type-2 probabilities lead to the same results as the usual ("type-1") probabilities: indeed, based on the type-2 probability, we can determine the probability of each event E – by computing the mean value of the probability $p(E|f)$ of this event over all possible distributions f: $p(E) = \int P(E|f)\,df$. This is true – just like it is true that we can reduce type-2 to type-1 and that we can defuzzify type-1 into an exact number – but in the fuzzy case, it does not means that each fuzzy number is equal to its defuzzified value and that each type-2 fuzzy number is equal to its reduced type-1 version.

Similarly in the probabilistic case: we can show that type-2 probabilities are different; see, e.g., [21]. Let us explain why type-2 probability case is different from the traditional ("type-1") probability case.

In the traditional case, we assume that all measuring instruments have the same probability distribution of measurement errors, and thus, the same variance σ^2. In this case, if we select any computable or random subsequence of the sequence of measurement errors, then for this subsequence, the population variance will be the same – close to σ^2.

On the other hand, type-2 probability means that different measuring instruments have slightly different variances $\sigma_1^2, \ldots, \sigma_n^2$. In this case, if we take into account all the measurements together, then for the resulting joint population, the variance is equal to

$$\sigma^2 = p_1 \cdot \sigma_1^2 + \ldots + p_n \cdot \sigma_n^2,$$

where p_i denotes the frequency with which we use the i-th instrument. However, in this case, if we select a (clearly computable) subsequence of all the measurement performed by the i-th instrument, then for this subsequence, the sample variance will be close to σ_i^2 – a different number.

Type-2 probabilities have been successfully used in practice: a practical example. Of course, Bayesian approach has been successfully used in many practical applications – but proponents of the more traditional "frequentist" approach claim that the difference is purely philosophical, and that the results of their approach are equally good. But there are some cases when empirically, the type-2 probabilistic approach leads to better results. One of these cases is described in [23], where it leads to a better description of spatial resolution of the results of seismic data processing.

Let us briefly describe the correpsonding practical problem. The ultimate goals of seismic data processing are:

- to find underground mineral deposits and
- to find potential underground earthquake-generating faults.

To achieve each of these goals, we trace how seismic signals propagate between nearby points on the surface:

- these can be signals from the actual distance earthquakes,
- these can be artificial signals generated by small exploratory explosions or by vibrating the soil.

In all the cases, we only get an approximate locations of the zones of interest, be it mineral zones or fault zones. Traditionally, probabilistic methods are used to gauge the location accuracy. It turns out that if we take into account that these probabilities are only approximately known – i.e., in effect, if we use type-2 probabilistic approach – we get more precise estimates of the resulting location accuracy.

Why type-2 probabilities lead to a more precise description of uncertainty. According to statistics (see, e.g., [37]), if we have n measurement $\widetilde{x}_i$ of the same quantity x performed with different standard deviations σ_i, then the most accurate combination of these measurements is

$$\widetilde{x} = \frac{\widetilde{x}_1 \cdot \sigma_1^{-2} + \ldots + \widetilde{x}_1 \cdot \sigma_n^{-2}}{\sigma_1^{-2} + \ldots + \sigma_n^{-2}},$$

and the standard deviation of the difference $\widetilde{x} - x$ is equal to

$$\sigma_{(2)}^2 = \frac{1}{\sigma_1^{-2} + \ldots + \sigma_n^{-2}}.$$

If we ignore the difference between these standard deviations – i.e., ignore the type-2 character of the situation – and assume that all the measurements are performed with the same average accuracy

$$\sigma^2 = \frac{\sigma_1^2 + \ldots + \sigma_n^2}{n},$$

then

$$\sigma^{-2} = \frac{n}{\sigma_1^2 + \ldots + \sigma_n^2},$$

and we can only reach the accuracy

$$\sigma_{(1)}^2 = \frac{1}{\sigma^{-2} + \ldots + \sigma^{-2}} = \frac{1}{n \cdot \sigma^{-2}} = \frac{\sigma_1^2 + \ldots + \sigma_n^2}{n^2}.$$

One can show that we always have $\sigma_{(1)}^2 < \sigma_{(2)}^2$ – i.e., that, by using the type-1 probabilistic approach, we under-estimate uncertainty.

Such an underestimations may be disastrous for safety-critical applications like earthquake safety studies:

- based on the small error estimate, we may think that the buildings are within the safe margins,
- but in reality, they may be prone to be damaged by a future earthquake.

How to prove that type-1 probabilistic computations under-estimate uncertainty. To prove that $\sigma_{(1)}^2 < \sigma_{(2)}^2$, we can use the known geometric fact that the

scalar (dot) product $x \cdot y = x_1 \cdot y_1 + \ldots + x_n \cdot y_n$ of two vectors $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$ is always smaller than or equal to the product $\|x\| \cdot \|y\|$ of their lengths $\|x\| = \sqrt{x_1^2 + \ldots + x_n^2}$ and $\|y\| = \sqrt{y_1^2 + \ldots + y_n^2}$, and the dot product $x \cdot y$ is only equal to the product of the lengths when the vectors differ from other by some multiplicative constant c: $y_i = c \cdot x_i$ for all i. In particular, for $x_i = \sigma_i$ and $y_i = \sigma_i^{-1}$, we have

$$x \cdot y = \sigma_1^2 \cdot \sigma_1^{-2} + \ldots + \sigma_n^2 \cdot \sigma_n^{-2} = 1 + \ldots + 1 = n,$$

so the above inequality takes the form

$$n \le \sqrt{\sigma_1^2 + \ldots + \sigma_n^2} \cdot \sqrt{\sigma_1^{-2} + \ldots + \sigma_n^{-2}}.$$

By taking squares of both sides, we conclude that

$$n^2 \le (\sigma_1^2 + \ldots + \sigma_n^2) \cdot (\sigma_1^{-2} + \ldots + \sigma_n^{-2}).$$

Dividing both side by the second sum and by n^2, we conclude that

$$\frac{1}{\sigma_1^{-2} + \ldots + \sigma_n^{-2}} \le \frac{\sigma_1^2 + \ldots + \sigma_n^2}{n^2},$$

i.e., exactly that $\sigma_{(1)}^2 \le \sigma_{(2)}^2$.

The only case when we have equality is when $y_i = c \cdot x_i$ for some c, i.e., in this case, when $\sigma_i^{-i} = c \cdot \sigma_i$ for all i. Multiplying both sides of this equality by σ_i, we conclude that in the case of equality, we have $\sigma_i^2 = c$, i.e., $\sigma_i = \sqrt{c}$ for all i – when all standard deviations are equal.

So indeed, if some measuring instruments are different, we can conclude that $\sigma_{(1)}^2 < \sigma_{(2)}^2$, i.e., that type-1 techniques under-estimate the uncertainty of the result.

There is another reason why we need to go beyond probabilities. In our description of how we get the probability distribution, we used the measurement errors Δx, i.e., the difference between the measurement result $\widetilde{x}$ and the actual value x. Of course, in reality, we do not know the actual value – but in many cases, we can measure the same quantity by a much more accurate measuring instrument; such measuring instrument is known as *standard*. The measurement error of the standard measuring instrument is much smaller than the measurement error of the original measurement; thus, we can safely ignore this much smaller measurement error, and safely assume that the results of the standard measurement instrument are the actual values.

Sometimes, this is possible, but sometimes, we already use state-of-the-art measuring instruments – for which no more accurate instrument is known. This happens in fundamental science, this happens in oil and gas industry, where it is cheaper to buy the most expensive state-of-the-art measuring instrument than to risk big losses when we end up drilling in a location that does not have mineral resources.

In other situations, standard measuring instruments exist, but practitioners do not use them – because they are expensive to use. For example, many sensors are

now very cheap: kids buy them on their own money and use them to build robots. However, more accurate measuring instruments are still expensive to use – so in manufacturing, when the profit margin is small, companies do not determine the probability distributions of the measurement errors. What can we do in such situations?

Enter interval uncertainty. By definition, a measuring instrument must guarantee something – otherwise, if its measurement does not provide any reliable information about the actual value, it would not be called a measuring instrument. So, the manufacturer of the measuring instrument must at least provide some upper bound Δ on the measurement accuracy: $|\Delta x| \le \Delta$. Indeed, if no such bound is provided, then, no matter what is the measurement result, the actual value can be arbitrarily large.

Once such a bound Δ is provided, then, based on the measurement result $\widetilde{x}$, we can conclude that the actual value x of the measured quantity lies somewhere in the interval $[\underline{x},\overline{x}] = [\widetilde{x}-\Delta,\widetilde{x}+\Delta]$. This case is known as *interval uncertainty*; see, e.g., [15, 26, 28, 31].

From the usual interval uncertainty to type-2 interval uncertainty. Different measuring instruments are slight different. In particular, different measuring instrument may have different values of the largest absolute value $|\Delta x|$. In addition to the upper bound Δ on such values, it makes sense to also provide a lower bound $\underline{\Delta}$ – to let the users know how much they can gain if they calibrate individual sensors instead of just relying on the general upper bound. In this case, instead of exact endpoints $\underline{x}$ and $\overline{x}$ for the interval containing x, we have intervals of possible values:

- the interval $[\widetilde{x}-\Delta,\widetilde{x}-\underline{\Delta}]$ of possible values of $\underline{x}$, and
- the interval $[\widetilde{x}+\underline{\Delta},\widetilde{x}+\Delta]$ of possible values of $\overline{x}$.

This case is similar to type-2 fuzzy, so it can be considered as type-2 intervals; see, e.g., [21].

Acknowledgments

This work was supported by the 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.

References

1. R. A. Alief, O. H. Huseynov, R. R. Aliyev, and A. A. Alizadeh, *The Arithmetic of Z-Numbers: Theory and Applications*, World Scientific, Singapore, 2015.
2. R. A. Aliev and V. Kreinovich, "Z-Numbers and Type-2 Fuzzy Sets: A Representation Result", *Intelligent Automation and Soft Computing*, 2018, Vol. 24, No. 1, pp. 205–210.
3. R. Belohlavek, J. W. Dauben, and G. J. Klir, *Fuzzy Logic and Mathematics: A Historical Perspective*, Oxford University Press, New York, 2017.
4. L. Bokati, A. Velasco, and V. Kreinovich, "Scale-Invariance and Fuzzy Techniques Explain the Empirical Success of Inverse Distance Weighting and of Dual Inverse Distance Weighting in Geosciences", *Proceedings of the Annual Conference of the North American Fuzzy Information Processing Society NAFIPS'2020*, Redmond, Washington, August 20–22, 2020, pp. 379–390.
5. O. Castillo, J. Castro, and P. Melin, *Interval Type-3 Fuzzy Systems: Theory and Design*, Springer, Cham, Switzerland, 2022.
6. O. Castillo, J. Castro, and P. Melin, "A methodology for building interval type-3 fuzzy systems based on the principle of justifiable granularity", *International Journal of Intelligent Systems*, 2022, doi 10.1002/int.22910
7. O. Castillo, J. Castro, and P. Melin, "Interval type-3 fuzzy aggregation of neural networks for multiple time series prediction: the case of financial forecasting", *Axioms*, 2022, Vol. 11, Paper 251.
8. O. Castillo, J. Castro, and P. Melin, "Interval type-3 fuzzy control for automated tuning of image quality in televisions", *Axioms*, 2022, Vol. 11, Paper 276.
9. F. Cervantes, B. Usevitch, L. Valera, and V. Kreinovich, "Why Sparse? Fuzzy Techniques Explain Empirical Efficiency of Sparsity-Based Data- and Image-Processing Algorithms", *Proceedings of the 2016 World Conference on Soft Computing*, Berkeley, California, May 22–25, 2016, pp. 165–169.
10. F. Cervantes, B. Usevitch, L. Valera, and V. Kreinovich, "Why Sparse? Fuzzy Techniques Explain Empirical Efficiency of Sparsity-Based Data- and Image-Processing Algorithms", In: L. Zadeh, R. R. Yager, S. N. Shahbazova, M. Reformat, and V. Kreinovich (eds.), *Recent Developments and New Direction in Soft Computing: Foundations and Applications*, Springer Verlag, Cham, Switzerland, 2018, pp. 419–430.
11. F. Cervantes, B. Usevitch, L. Valera, V. Kreinovich, and O. Kosheleva, "Fuzzy Techniques Provide a Theoretical Explanation for the Heuristic l^p-Regularization of Signals and Images", *Proceedings of the 2016 IEEE International Conference on Fuzzy Systems FUZZ-IEEE'2016*, Vancouver, Canada, July 24–29, 2016.
12. S. Ferson, V. Kreinovich, L. Ginzburg, D. S. Myers, and K. Sentz, *Constructing Probability Boxes and Dempster-Shafer Structures*, Sandia National Laboratories, Report SAND2002-4015, January 2003.
13. R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics*, Addison Wesley, Boston, Massachusetts, 2005.
14. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, Cambridge, Massachusetts, 2016.
15. L. Jaulin, M. Kiefer, O. Didrit, and E. Walter, *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control, and Robotics*, Springer, London, 2012.
16. E. T. Jaynes and G. L. Bretthorst, *Probability Theory: The Logic of Science*, Cambridge University Press, Cambridge, UK, 2003.
17. G. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic*, Prentice Hall, Upper Saddle River, New Jersey, 1995.
18. O. Kosheleva, V. Kreinovich, and S. Shahbazova, "Type-2 Fuzzy Analysis Explains Ubiquity of Triangular and Trapezoid Membership Functions", In: S. N. Shahbazova, J. Kacprzyk, V. E. Balas, and V. Kreinovich (eds.), *Recent Developments and the New Direction in Soft-Computing Foundations and Applications: Selected Papers from the World Conference on Soft Computing, Baku, Azerbaijan, May 29–31, 2018*, Springer, Cham, Switzerland, 2021, pp. 63–75.

19. V. Kreinovich, "From Processing Interval-Valued Fuzzy Data to General Type-2: Towards Fast Algorithms", *Proceedings of the IEEE Symposium on Advances in Type-2 Fuzzy Logic Systems T2FUZZ'2011*, part of the *IEEE Symposium Series on Computational Intelligence*, Paris, France, April 11–15, 2011, pp. ix–xii.
20. V. Kreinovich and O. Kosheleva, "Fuzzy Or Neural, Type-1 Or Type-2 – When Each Is Better: First-Approximation Analysis", In: Y. P. Kondratenko, V. Kreinovich, W. Pedrycz, A. A. Chikrii, and A. M. Gil Lafuente (eds.), *Artificial Intelligence in Control and Decision-Making Systems*, Springer, 2023, pp. 67–74.
21. V. Kreinovich, O. Kosheleva, and L. Longpré, "From Type-2 Fuzzy to Type-2 Intervals and Type-2 Probabilities", In: K. T. Atanassov, V. Atanassova, J. Kacprzyk, A. Kaluszko, M. Krawczak, J. Owsinski, S. N. Sotirov, E. Sotirova, E. Szmidt, and S. Zadrozny (eds.), *Uncertainty and Imprecision in Decision Making and Decision Support – New Advances, Challenges, and Perspectives. Selected Papers from the IWIFSGN-2023 – Twenty First International Workshop on Intuitionistic Fuzzy Sets and Generalized Nets, Warsaw, Poland, October 20–23, 2023*, Lecture Notes in Networks and Systems Vol. 1550, Springer, Cham, Switzerland, 2026, pp. 13–22.
22. V. Kreinovich, O. Kosheleva, P. Melin, and O. Castillo, "Efficient algorithms for data processing under type-3 (and higher) fuzzy uncertainty", *Mathematics*, 2022, Vol. 10, Paper 2361.
23. V. Kreinovich, J. Nava, R. Romero, J. Olaya, A. Velasco, and K. C. Miller, "Spatial Resolution for Processing Seismic Data: Type-2 Methods for Finding the Relevant Granular Structure", *Proceedings of the IEEE International Conference on Granular Computing GrC'2010*, Silicon Valley, USA, August 14–16, 2010.
24. V. Kreinovich and C. D. Stylios, "When Should We Switch from Interval-Valued Fuzzy to Full Type-2 Fuzzy (e.g., Gaussian)?", *Critical Review*, 2015, Vol. XI, pp. 57–66.
25. V. Kreinovich and G. Xiang, "Towards Fast Algorithms for Processing Type-2 Fuzzy Data: Extending Mendel's Algorithms From Interval-Valued to a More General Case", Proceedings of the 27th International Conference of the North American Fuzzy Information Processing Society NAFIPS'2008, New York, New York, May 19-22, 2008.
26. B. J. Kubica, *Interval Methods for Solving Nonlinear Constraint Satisfaction, Optimization, and Similar Problems: from Inequalities Systems to Game Solutions*, Springer, Cham, Switzerland, 2019.
27. J. Lorkowski, R. Aliev, and V. Kreinovich, "Towards Decision Making under Interval, Set-Valued, Fuzzy, and Z-Number Uncertainty: A Fair Price Approach", *Proceedings of the IEEE World Congress on Computational Intelligence WCCI'2014*, Beijing, China, July 6–11, 2014.
28. G. Mayer, *Interval Analysis and Automatic Result Verification*, de Gruyter, Berlin, 2017.
29. J. M. Mendel, *Explainable Uncertain Rule-Based Fuzzy Systems*, Springer, Cham, Switzerland, 2024.
30. G. A. Miller, "The magical number seven plus or minus two: some limits on our capacity for processing information", *Psychological Review*, 1956, Vol. 63, No. 2, pp. 81–97.
31. R. E. Moore, R. B. Kearfott, and M. J. Cloud, *Introduction to Interval Analysis*, SIAM, Philadelphia, 2009.
32. H. T. Nguyen, C. L. Walker, and E. A. Walker, *A First Course in Fuzzy Logic*, Chapman and Hall/CRC, Boca Raton, Florida, 2019.
33. V. Novák, I. Perfilieva, and J. Močkoř, *Mathematical Principles of Fuzzy Logic*, Kluwer, Boston, Dordrecht, 1999.
34. S. G. Rabinovich, *Measurement Errors and Uncertainty: Theory and Practice*, Springer Verlag, New York, 2005.
35. S. K. Reed, *Cognition: Theories and Application*, SAGE Publications, Thousand Oaks, California, 2022.
36. C. Servin, O. Kosheleva, and V. Kreinovich, "Why Interval-Valued (and Type-2) Fuzzy Methods Are Often More Effective", In: M. Z. Reformat, S. Senatore, Y. Nojima, and V. Kreinovich (eds.), *Fuzzy Systems 60 Years Later: Past, Present, and Future. Proceedings of the Joint 2025 World Congress on the International Fuzzy Systems Association and the 2025 Annual Conference of the North American Fuzzy Information Processing Society IFSA-NAFIPS 2025, Banff, Canada, August 16–19, 2025*, Springer, Cham, Switzerland.

37. D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman and Hall/CRC, Boca Raton, Florida, 2011.
38. K. S. Thorne and R. D. Blandford, *Modern Classical Physics: Optics, Fluids, Plasmas, Elasticity, Relativity, and Statistical Physics*, Princeton University Press, Princeton, New Jersey, 2021.
39. L. A. Zadeh, "Fuzzy sets", *Information and Control*, 1965, Vol. 8, pp. 338–353.
40. L. A. Zadeh, "A Note on Z-Numbers", *Information Sciences*, 2011, Vol. 181, pp. 2923–2932.