Relations between Interval Computing and Soft Computing

Vladik Kreinovich

Department of Computer Science, University of Texas at El Paso, 500 W. University, El Paso, TX 79968, USA vladik@utep.edu

This volume is about knowledge processing with interval and soft computing, i.e., about techniques that use both interval and soft computing to process knowledge and about the results of applying these techniques. To better understand these techniques, in Chapter 1, we described fundamentals of interval computing, and in Chapter 2, we described the fundamentals of soft computing. Now it is time to explain how these techniques are related – and how they can be combined. Some examples of such a relation were already given in Chapter 2 – e.g., interval-valued fuzzy sets. Now it is time to provide a systematic description of this relation. After this chapter, we will be ready to describe how to combine interval and fuzzy techniques, and how the resulting combined techniques can be applied to real-life problems.

This chapter starts with a brief reminder of why data processing and knowledge processing are needed in the first place, why interval and fuzzy methods are needed for data and knowledge processing, and which of the possible data and knowledge processing techniques we should use. Then, we explain how these reasonable soft computing techniques are naturally related with interval computing. Finally, we explain the need for interval-valued fuzzy techniques – techniques which will be used a lot in our future applications – and how the transition to such techniques is also related to interval computing.

1 Why Data Processing and Knowledge Processing Are Needed in the First Place: A Brief Reminder

1.1 Classification of Practical Problems

Most practical problems can be crudely classified into three classes:

- we want to *learn* what is happening in the world; in particular, we want to know the numerical values of different quantities (distances, masses, charges, coordinates, etc.);

- based on these values, we would like to *predict* how the state of the world will change over time;
- finally, we would like to find out what *changes* we need to make in the world so that these changes will lead to the desired results.

A real-life problem often involves solving subproblems of all three types.

This classification is closely related to the well-known classification of practically useful creative activity into engineering and science:

- The tasks of learning the current state of the world and predicting the future state of the world are usually classified as *science*.
- The tasks of finding the appropriate change are usually classified as *engineering*.

For example, measuring the flow of the Rio Grande river at different locations and predicting how this river flow will change over time are problems of science. Finding the best way to change this flow (e.g., by building a levee to protect downtown El Paso) is a problem of engineering.

1.2 First Class of Practical Problems: Learning the State of the World

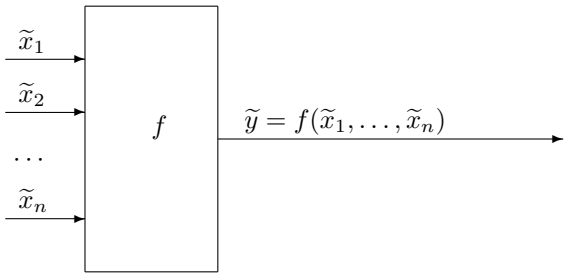
Let us start with the first class of practical problems: the problem of learning the state of the world. As we have mentioned, this means, in particular, that we want to know the numerical values of different quantities y that characterize this state.

Some quantities y we can simply directly measure. For example, when we want to know the current state of a patient in a hospital, we can measure the patient's body temperature, blood pressure, weight, and many other important characteristics. In some situations, we do not even need to measure: we can simply ask an expert, and the expert will provide us with an (approximate) value $\widetilde{y}$ of the quantity y.

However, many other quantities of interest are difficult or even important to measure or estimate directly. Examples of such quantities include the amount of oil in a given well or a distance to a star. Since we cannot directly measure the values of these quantities, the only way to learn some information about them is: to measure (or ask an expert to estimate) some other easier-to-measure quantities $x_1, \ldots, x_n$, then to estimate y based on the measured values $\widetilde{x}_i$ of these auxiliary quantities x_i.

For example, to estimate the amount of oil in a given well, we perform *seismic* experiments: we set up small explosions at some locations and measure the resulting seismic waves at different distances from the location of the explosion. To find the distance to a faraway star, we measure the direction to the star from different locations on Earth (and/or in different seasons) and the coordinates of (and the distances between) the locations of the corresponding telescopes.

To estimate the value of the desired quantity y, we must know the relation between y and the easier-to-measure (or easier-to-estimate) quantities $x_1, \ldots, x_n$. Specifically, we want to use the estimates of x_i to come up with an estimate for y. Thus, the relation between y and x_i must be given in the form of an *algorithm* $f(x_1, \ldots, x_n)$ which transforms the values of x_i into an estimate for y. Once we know this algorithm f and the measured values $\widetilde{x}_i$ of the auxiliary quantities, we can estimate y as $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$.



In different practical situations, we have algorithms f of different complexity. For example, to find the distance to star, we can usually have an explicit analytical formula coming from geometry. In this case, f is a simple formula. On the other hand, to find the amount of oil, we must numerically solve a complex partial differential equation. In this case, f is a complex iterative algorithm for solving this equation.

In the case when the values x_i are obtained by measurement, this two-stage process does involve measurement. To distinguish it from *direct* measurements (i.e., measurements which directly measure the values of the desired quantity), the above two-stage process is called an *indirect* measurement.

1.3 Second Class of Practical Problems: Predicting the Future State of the World

Once we know the values of the quantities $y_1, \ldots, y_m$ which characterize the current state of the world, we can start predicting the future state of the world, i.e., the future values of these quantities.

To be able to predict the future value z of each of these quantities, we must know how exactly this value z depends on the current values $y_1, \ldots, y_m$. Specifically, we want to use the known estimates $\widetilde{y}_i$ for y_i to come up with an estimate for z. Thus, the relation between z and y_i must be given in the form of an *algorithm* $g(y_1, \ldots, y_m)$ which transforms the values of y_i into an estimate for z. Once we know this algorithm g and the estimates $\widetilde{y}_i$ for the current values of the quantities, we can estimate z as $\widetilde{z} = g(\widetilde{y}_1, \ldots, \widetilde{y}_n)$.

The corresponding algorithm g can be very complicated and time-consuming. This is, e.g., how weather is predicted now: weather prediction requires so many computations that it can only be performed on fast supercomputers.

1.4 The General Notion of Data and Knowledge Processing

So far, we have analyzed two different classes of practical problems:

- the problem of *learning* the current state of the world (i.e., the problem of indirect measurement), and
- the problem of *predicting* the future state of the world.

From the *practical* viewpoint, these two problems are drastically different. However, as we have seen, from the *computational* viewpoint, these two problems are very similar. In both problems,

- we start with the estimates $\widetilde{x}_1, \ldots, \widetilde{x}_n$ for the quantities $x_1, \ldots, x_n$, and then
- we apply the known algorithm f to these estimates, resulting in an estimate $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$ for the desired quantity y.

When the inputs come from measurements – i.e., constitute *data* – the computational part of the corresponding procedure is called *data processing*. When the inputs come from experts – i.e., constitute *knowledge* – the computational part of the corresponding procedure is called *knowledge processing*.

1.5 Third Class of Practical Problems: How to Change the World

Once we know the current state of the world and we know how to predict the consequences of different decisions (designs, etc.), it is desirable to find the decision (design, etc.) which guarantees the given results. Depending on what we want from this design, we can subdivide all the problems from this class into two subclasses. In both subclasses, the design must satisfy some constraints. Thus, we are interested in finding a design that satisfies all these constraints.

- In some practical situations, satisfaction of all these constraints is all we want. In general, there may be several possible designs which satisfy given constraints. In the problems from the first subclass, we do not have any preferences for one of these designs – any one of them will suffice. Such problems are called the problems of *constraint satisfaction*.
- In other practical situations, we do have a clear preference between different designs x. This preference is usually described in terms of an *objective function* $F(x)$ – a function for which more preferable designs x correspond to larger values of $F(x)$. In such situation, among all the designs which satisfy given constraints, we would like to find a design x for which the value $F(x)$ of the given objective function is the largest. Such problems are called *optimization problems*.

2 Need for Interval Computations

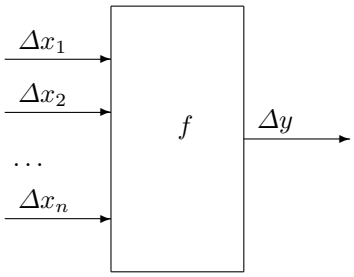
2.1 Need to Take Uncertainty Into Account

In the case of data processing, we start with measurement results $\widetilde{x}_1, \ldots, \widetilde{x}_n$. Measurements are never exact. There is a non-zero difference $\Delta x_i \stackrel{\text{def}}{=} \widetilde{x}_i - x_i$ between the (approximate) measurement result $\widetilde{x}_i$ and the (unknown) actual value x_i of the i-th quantity x_i. This difference is called the *measurement error*. The result $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$ of applying the algorithm f to the measurement results $\widetilde{x}_i$ is, in general, different from the result $y = f(x_1, \ldots, x_n)$ of applying this algorithm to the actual values x_i. Thus, our estimate $\widetilde{y}$ is, in general, different from the actual value y of the desired quantity: $\Delta y \stackrel{\text{def}}{=} \widetilde{y} - y \neq 0$.

In many practical applications, it is important to know not only the desired estimate for the quantity y, but also how accurate this estimate is. For example, in geophysical applications, it is not enough to know that the amount of oil in a given oil field is about 100 million tons: it is also important to know how accurate this estimate is. If the amount is 100 ± 10, this means that the estimates are good enough, and we should start exploring this oil field. On the other hand, if it is 100 ± 200, this means that it is quite possible that the actual value of the desired quantity y is 0, i.e., that there is no oil at all. In this case, it may be prudent to perform additional measurements before we invest a lot of money into drilling oil wells.

The situation becomes even more critical in medical emergencies: it is not enough to have an estimate of blood pressure or body temperature to make a decision (e.g., whether to perform a surgery), it is important that even with the measurement uncertainty, we are sure about the diagnosis – and if we are not, maybe it is desirable to perform more accurate measurements.

It is therefore desirable to find out the uncertainty Δy caused by the uncertainties Δx_i in the inputs:



Comment. We assumed that the relation f provides the *exact* relation between the variables $x_1, \ldots, x_n$, and the desired value y. If so, then, in the ideal case in which we plug in the actual (unknown) values of x_i into the algorithm f, we get the exact value $y = f(x_1, \ldots, x_n)$ of y.

In many real-life situations, the relation f between x_i and y is only *approximately* known. In this case, even if we know the exact values of x_i, substituting

these values into the approximate function f will not provide us with the exact value of y. In such situations, there is even more uncertainty in y:

- first, there is an uncertainty in y caused by the the uncertainty in the inputs;
- second, there is a *model uncertainty* caused by the fact that the known algorithm f only provides an approximate description of the dependence between the inputs and the output.

A model uncertainty has to be estimated separately and added to the uncertainty caused by the measurement errors.

2.2 From Probabilistic to Interval Uncertainty

To estimate the uncertainty Δy caused by the measurement uncertainties Δx_i, we need to have some information about these original uncertainties Δx_i. The whole idea of uncertainty is that we do not know the exact value of x_i (hence, we do not know the exact value of Δx_i). In order words, there are several possible values of Δx_i. Thus, the first thing we would like to know is the *set* of possible values of Δx_i.

We may also know that some of these possible values are more frequent than the others. In other words, we may also have some information about the *probabilities* of different possible values Δx_i.

The manufacturers of a measuring device usually provide us with an upper bound Δ_i for the (absolute value of) possible measurement errors, i.e., with the bound Δ_i for which we are guaranteed that $|\Delta x_i| \le \Delta_i$.

The need for such a bound comes from the very nature of a measurement process. Indeed, if no such bound is provided, this means that the actual value x_i can be as different from the "measurement result" $\widetilde{x}_i$ as possible. Such a value $\widetilde{x}_i$ is not a measurement, it is a wild guess.

Since the (absolute value of the) measurement error $\Delta x_i = \tilde{x}_i - x_i$ is bounded by the given bound Δ_i, we can therefore guarantee that the actual (unknown) value of the desired quantity belongs to the interval

$$\mathbf{x}_i \stackrel{\text{def}}{=} [\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i].$$

For example, if the measured value of a quantity is $\widetilde{x}_i = 1.0$, and the upper bound Δ_i on the measurement error is 0.1, this means that the (unknown) actual value of the measured quantity can be anywhere between $1 - 0.1 = 0.9$ and $1 + 0.1 = 1.1$, i.e., that it can take any value from the interval $[0.9, 1.1]$.

In many practical situations, we not only know the interval $[-\Delta_i, \Delta_i]$ of possible values of the measurement error; we also know the probability of different values Δx_i within this interval [8].

In most practical applications, it is assumed that the corresponding measurement errors are normally distributed with 0 mean and known standard

deviation. Numerous engineering techniques are known (and widely used) for processing this uncertainty; see, e.g., [8].

In practice, we can determine the desired probabilities of different values of Δx_i by comparing

- the result $\widetilde{x}_i$ of measuring a certain quantity with this instrument and
- the result $\widetilde{x}_{i\,\mathrm{st}}$ of measuring the same quantity by a standard (much more accurate) measuring instrument.

Since the standard measuring instrument is much more accurate than the one we use, i.e., $|\widetilde{x}_{i\,\mathrm{st}} - x_i| \ll |\widetilde{x}_i - x_i|$, we can assume that $\widetilde{x}_{i\,\mathrm{st}} = x_i$, and thus, that the difference $\widetilde{x}_i - \widetilde{x}_{i\,\mathrm{st}}$ between these two measurement results is practically equal to the measurement error $\Delta x_i = \widetilde{x}_i - x_i$. Thus, the empirical distribution of the difference $\widetilde{x}_i - \widetilde{x}_{i\,\mathrm{st}}$ is close to the desired probability distribution for measurement error.

There are two cases, however, when this determination is not done:

- First is the case of *cutting-edge* measurements, e.g., measurements in fundamental science. When the Hubble telescope detects the light from a distant galaxy, there is no "standard" (much more accurate) telescope floating nearby that we can use to calibrate the Hubble: the Hubble telescope is the best we have.
- The second case is the case of real *industrial* applications (such as measurements on the shop floor). In this case, in principle, every sensor can be thoroughly calibrated, but sensor calibration is so costly – usually costing several orders of magnitude more than the sensor itself – that manufacturers rarely do it (only if it is absolutely necessary).

In both cases, we have no information about the probabilities of Δx_i; the only information we have is the upper bound on the measurement error.

In such cases, after performing a measurement and getting a measurement result $\widetilde{x}_i$, the only information that we have about the actual value x_i of the measured quantity is that it belongs to the interval $\mathbf{x}_i = [\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i]$. In other words, we do know not the actual value x_i of the i-th quantity. Instead, we know the *interval* $[\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i]$ that contains x_i. In this situation, for each i, we know the interval $\mathbf{x}_i$ of possible values of x_i, and we need to find the range

$$\mathbf{y} \stackrel{\mathrm{def}}{=} \{f(x_1, \ldots, x_n) : x_1 \in \mathbf{x}_1, \ldots, x_n \in \mathbf{x}_n\}$$

of the given function $f(x_1, \ldots, x_n)$ over all possible tuples $x = (x_1, \ldots, x_n)$ with $x_i \in \mathbf{x}_i$. Since the function $f(x_1, \ldots, x_n)$ is usually continuous, this range is also an interval, i.e., $\mathbf{y} = [\underline{y}, \overline{y}]$ for some $\underline{y}$ and $\overline{y}$. So, to find this range, it is sufficient to find the endpoints $\underline{y}$ and $\overline{y}$ of this interval.

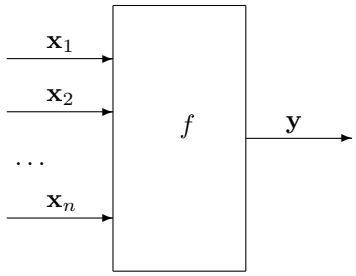
Let us formulate the corresponding *interval computations* problem in precise terms. We are given:

- an integer n;
- n intervals $\mathbf{x}_1 = [\underline{x}_1, \overline{x}_1]$, $\ldots$, $\mathbf{x}_n = [\underline{x}_n, \overline{x}_n]$, and

- an algorithm $f(x_1, \ldots, x_n)$ which transforms n real numbers into a real number $y = f(x_1, \ldots, x_n)$.

We need to compute the endpoints $\underline{y}$ and $\overline{y}$ of the interval

$$\mathbf{y} = [\underline{y}, \overline{y}] = \{f(x_1, \ldots, x_n) : x_1 \in [\underline{x}_1, \overline{x}_1], \ldots, [\underline{x}_n, \overline{x}_n]\}.$$



Interval computations are also important for the second class of problems: predicting future values.

3 Knowledge Processing and Fuzzy Uncertainty

3.1 Need to Process Fuzzy Uncertainty

In many practical situations, we only have expert estimates for the inputs x_i. Sometimes, experts provide guaranteed bounds on the x_i, and even the probabilities of different values within these bounds. However, such cases are rare. Usually, the experts' opinions about the uncertainty of their estimates are described by (imprecise, "fuzzy") words from natural language. For example, an expert can say that the value x_i of the i-th quantity is approximately equal to 1.0, with an accuracy most probably of about 0.1. Based on such "fuzzy" information, what can we say about $y = f(x_1, \ldots, x_n)$?

The need to process such "fuzzy" information was first emphasized in the early 1960s by L. Zadeh who designed a special technique of *fuzzy logic* for such processing; see, e.g., [1, 7].

3.2 Processing Fuzzy Uncertainty: Main Idea

Intuitively, a value y is a reasonable value of the desired quantity if $y = f(x_1, \ldots, x_n)$ for some reasonable values x_i, i.e., if for some values $x_1, \ldots, x_n$, x_1 is reasonable, x_2 is reasonable, ..., and $y = f(x_1 \ldots, x_n)$. Thus, to describe to what extent different values of y are reasonable, we must be able:

- to describe to what extent (to what degree) different values of x_i are reasonable, and
- to combine these degrees into the desired degree of belief in reasonability of y.

3.3 Degrees of Belief

Let us first introduce the basic concept of degrees of belief. For example, we would like to estimate to what extent the value $x_i = 0.89$ is consistent with the statement "the value x_i of the i-th quantity is approximately equal to 1.0, with an accuracy most probably about 0.1".

In the absence of uncertainty, every statement is either true or false. In the computer, "true" is usually represented as 1, and "false" as 0. It is therefore reasonable to use numbers between 0 and 1 to represent levels of confidence which are intermediate – intermediate between the absolute confidence that a given statement is true and the absolute confidence that a given statement is false.

How do we determine this degree of confidence? Some methods have been described in the previous chapter. For example, we can ask several (N) experts whether $x_i = 0.89$ is consistent with the above statement, and if M of them reply "yes", take the ratio M/N as the desired degree of confidence. If we do not have access to numerous experts, we can simply ask the only available expert to describe his or her degree of confidence by marking a number on a scale from 0 to N (e.g., on a scale from 0 to 5). If an expert marks his or her degree as M, we take the ratio M/N as the desired degree of confidence.

3.4 Membership Functions

To formally describe the original expert's statement S about x_i, we need to know, for every real number x_i, the degree $\mu_S(x_i)$ to which this real number is consistent with this statement S.

By using the above procedure, we can determine this value $\mu_S(x_i)$ for every given real number x_i. This procedure includes asking questions to the expert. In practice, we can only ask finitely many questions. Thus, no matter how many questions we ask, by using the above procedure, we can only find the values $\mu_S(x_i)$ for finitely many real numbers x_i. To estimate the values $\mu_S(x_i)$ for all other real numbers x_i, we must therefore use interpolation and extrapolation. As we have mentioned in the previous chapter, usually, a piecewise interpolation is used, but sometimes a more sophisticated procedure is applied: e.g., a piecewise quadratic interpolation.

The function $\mu_S(x_i)$ which is obtained by this approximation is called a *membership function*. This function describes, for every real number x_i, the degree $\mu_S(x_i)$ to which this real number is consistent with this statement S.

3.5 Need for "And"- and "Or"-Operations: t-Norms and t-Conorms

As we have mentioned earlier, we are not directly interested in the degree to which a given real number x_i is consistent with the expert's knowledge S_i about the i-th input. We are mainly interested in the degree to which x_1 is

consistent with the knowledge about the first input *and* x_2 is consistent with the knowledge about the second input *and* ... *and* x_n is consistent with the knowledge about the n-th input.

In principle, we can determine the degree of belief in such a composite statement by asking an expert, for each possible combination of values $x_1, x_2, \ldots, x_n$, what is the degree to which this combination is consistent with all the available expert knowledge. However, as we have mentioned earlier, even for a single input, we cannot realistically elicit degrees of confidence about too many values. If we consider N possible values of each input, then we would need to elicit the expert's degree of confidence about $N^n \gg N$ possible combinations – which is even less realistic.

Since we cannot directly elicit the expert's degree of confidence in all composite statements, a natural idea is to estimate the degree of confidence in the composite statement based on the degrees of confidence in individual statements – such as "x_i is consistent with the expert's knowledge S_i about the i-th input."

How can we come up with such an estimate? Let us reformulate this estimation problem:

- we know the expert's degree of confidence in statements A_1, A_2, ..., A_n, and
- we want to estimate the expert's degree of confidence in a composite statement $A_1 \,\&\, A_2 \,\&\, \ldots \,\&\, A_n$ (i.e., "A_1 and A_2 and ... and A_n").

Since, e.g., $A_1 \,\&\, A_2 \,\&\, A_3$ can be represented as $(A_1 \,\&\, A_2) \,\&\, A_3$, it is sufficient to solve this estimation problem for the case of two statements. Once we have a solution for this particular case, we will then be able to solve the general problem as well:

- first, we apply the two-statement solution to the degrees of certainty in A_1 and A_2, and get an estimate for the expert's degree of certainty in $A_1 \,\&\, A_2$;
- then, we apply the same solution to the degrees of certainty in $A_1 \,\&\, A_2$ and A_3, and get an estimate for the expert's degree of certainty in $A_1 \,\&\, A_2 \,\&\, A_3$;
- after that, we apply the same solution to the degrees of certainty in $A_1 \,\&\, A_2 \,\&\, A_3$ and A_4, and get an estimate for the expert's degree of certainty in $A_1 \,\&\, A_2 \,\&\, A_3 \,\&\, A_4$;
- etc.

Eventually, we will get the degree of confidence in the desired composite statement $A_1 \,\&\, A_2 \,\&\, \ldots \,\&\, A_n$.

Thus, we need a procedure that would transform the degree of belief d_1 in a statement A_1 and the degree of belief d_2 in a statement A_2 into a (reasonable) estimate for a degree of belief in a composite statement $A_1 \,\&\, A_2$. Let us denote the estimate corresponding to given values d_1 and d_2 by $f_{\&}(d_1, d_2)$. The procedure $f_{\&}$ that maps degrees of belief d_1 and d_2 in statements A_1 and A_2

into a degree of belief $d = f_{\&}(d_1, d_2)$ in $A_1 \,\&\, A_2$ is called an *"and"-operation*, or, for historical reasons, a *t-norm*.

Similarly, to estimate the degree of belief in a composite statement $A_1 \vee A_2$ ("A_1 or A_2"), we need a procedure $f_\vee$ that maps degrees of belief d_1 and d_2 in statements A_1 and A_2 into a degree of belief $d = f_\vee(d_1, d_2)$ in $A_1 \vee A_2$. Such a procedure is called an *"or"-operation* . Since in logic, "or" is a kind of dual to "and", an "or"-operation can be viewed as a dual to an "and"-operation (t-norm). Because of this duality, an "or"-operation is also called a *t-conorm*.

3.6 Properties of "And"- and "Or"-Operations

From the intended meaning of the "and"- and "or"-operations, we can deduce reasonable properties of these operations.

For example, intuitively, "A_1 and A_2" means the same as "A_2 and A_1". Thus, it is reasonable to require that our estimate $f_{\&}(d_1, d_2)$ for the degree of confidence in "A_1 and A_2" should be the same our estimate $f_{\&}(d_2, d_1)$ for the degree of confidence in "A_2 and A_1". In other words, we must have $f_{\&}(d_1, d_2) = f_{\&}(d_2, d_1)$ for all possible values of d_1 and d_2. In mathematical terms, this means that the function $f_{\&}$ must be *commutative*.

Similarly, "(A_1 and A_2) and A_3" means the same as "A_1 and (A_2 and A_3)" – because both mean the same as "A_1 and A_2 and A_3". For each "and"-operation $f_{\&}$, the expression "(A_1 and A_2) and A_3" means that we:

- first estimate the degree of belief in "A_1 and A_2" as $f_{\&}(d_1, d_2)$, and
- then estimate the degree of belief in "(A_1 and A_2) and A_3" as

$$f_{\&}(f_{\&}(d_1, d_2), d_3).$$

Similarly, the expression "A_1 and (A_2 and A_3)" means that we:

- first estimate the degree of belief in "A_2 and A_3" as $f_{\&}(d_2, d_3)$, and
- then estimate the degree of belief in "A_1 and (A_2 and A_3)" as

$$f_{\&}(d_1, f_{\&}(d_2, d_3)).$$

Since the expressions are equivalent, it is reasonable to require that these estimates coincide, i.e., that $f_{\&}(f_{\&}(d_1, d_2), d_3) = f_{\&}(d_1, f_{\&}(d_2, d_3))$ for all possible values of d_1, d_2, and d_3. In mathematical terms, this means that the function $f_{\&}$ must be *associative*.

There are several other reasonable properties of "and"-operations. For example, since "A_1 and A_2" implies A_1, our degree of belief in the composite statement "A_1 and A_2" cannot exceed our degree of belief in A_1. Thus, it is reasonable to require that the estimate $f_{\&}(d_1, d_2)$ for this degree of belief should also not exceed our degree of belief d_1 in the statement A_1. In other words, we should have $f_{\&}(d_1, d_2) \le d_1$ for all possible values of d_1 and d_2.

If A_1 is absolutely true (i.e., $d_1 = 1$), then intuitively, the composite statement "A_1 and A_2" has exactly the same truth value as A_2. Thus, it is reasonable to require that $f_{\&}(1, d_2) = d_2$ for all possible values of d_2.

On the other hand, if A_1 is absolutely false (i.e., $d_1 = 0$), then the composite statement "A_1 and A_2" should also be absolutely false, no matter how much we may believe in A_2. Thus, it is reasonable to require that $f_{\&}(0, d_2) = 0$ for all possible values of d_2.

Finally, if, due to a new evidence, our degree of belief in one of the statements A_1 and A_2 increases, the resulting degree of belief in a composite statement "A_1 and A_2" will either increase or stay the same – but it cannot decrease. Thus, it is it is reasonable to require that the operation $f_{\&}$ be *monotonic* in the sense that if $d_1 \le d_1'$ and $d_2 \le d_2'$, then $f_{\&}(d_1, d_2) \le f_{\&}(d_1', d_2')$.

All these properties are indeed required of an "and"-operation (t-norm). Similarly, it is reasonable to require that an "or"-operation (t-conorm) $f_\vee$ should be commutative, associative, monotonic, and satisfy the conditions that $d_1 \le f_\vee(d_1, d_2)$, $f_\vee(1, d_2) = 1$, and $f_\vee(0, d_2) = d_2$ for all possible values of d_1 and d_2.

3.7 Simplest "And"- and "Or"-Operations: Derivation

There exist many different "and"- and "or"-operations which satisfy the above properties; see, e.g., [1, 4, 5, 7]. In some applications such as fuzzy control (see Chapter 2), it is crucial to select appropriate operations – because we can use the corresponding additional degrees of freedom to tune the resulting control and make it an even better fit for the corresponding objective function.

However, in knowledge processing, when we are very uncertain about the inputs, it is probably more reasonable to select the simplest "and"- and "or"-operations which are consistent with the expert knowledge. To select such operations, it makes sense to consider yet another property of "and" and "or" – that for every statement A, "A and A" means the same as simply A. Thus, it is reasonable to require that for every statement A with a degree of confidence d, our estimate $f_{\&}(d, d)$ of the expert's degree of confidence in "A and A" should be the same as the original degree of confidence d in the original statement A. Thus, it is reasonable to require that $f_{\&}(d, d) = d$ for all possible values of d. In mathematical terms, this means that the function $f_{\&}$ must be *idempotent*.

Similarly, since "A or A" means the same as simply A, it is reasonable to require that $f_\vee(d, d) = d$ for all possible values of d, i.e., that the function $f_{\&}$ must also be *idempotent*.

It turns out that this additional requirement leads to a unique "and"-operation and a unique "or"-operation. Let us first show that the only idempotent "and"-operation is $f_{\&}(d_1, d_2) = \min(d_1, d_2)$. Without loss of generality, let us assume that $d_1 \le d_2$. In this case, the desired equality takes the form $f_{\&}(d_1, d_2) = d_1$. Since the operation $f_{\&}$ is idempotent, we have $f_{\&}(d_1, d_1) = d_1$. Due to $d_1 \le d_2$, monotonicity implies that $f_{\&}(d_1, d_1) \le f_{\&}(d_1, d_2)$, hence $d_1 \le f_{\&}(d_1, d_2)$. On the other hand, for an "and"-operation, we always have $f_{\&}(d_1, d_2) \le d_1$. So, we can conclude that $f_{\&}(d_1, d_2) = d_1$, i.e., indeed, $f_{\&}(d_1, d_2) = \min(d_1, d_2)$.

Let us now prove that the only idempotent "or"-operation is $f_\vee(d_1, d_2) = \max(d_1, d_2)$. Without loss of generality, let us again assume that $d_1 \le d_2$. In this case, the desired equality takes the form $f_\vee(d_1, d_2) = d_2$. Since the operation $f_\vee$ is idempotent, we have $f_\vee(d_2, d_2) = d_2$. Due to $d_1 \le d_2$, monotonicity implies that $f_\vee(d_1, d_2) \le f_\vee(d_2, d_2)$, hence $f_\vee(d_1, d_2) \le d_2$. On the other hand, for an "or"-operation, we always have $d_2 \le f_\vee(d_1, d_2)$. Thus, we conclude that $f_\vee(d_1, d_2) = d_2$, i.e., indeed, $f_\vee(d_1, d_2) = \max(d_1, d_2)$.

The operations $f_\&(d_1, d_2) = \min(d_1, d_2)$ and $f_\vee(d_1, d_2) = \max(d_1, d_2)$ were actually the first designed by L. Zadeh; they are still actively used in various applications of fuzzy techniques; see, e.g., [1, 7].

3.8 Zadeh's Extension Principle

Let us apply the above simple operations to knowledge processing, or, to be more precise, to processing fuzzy uncertainty. In this situation:

- We know an algorithm $y = f(x_1, \ldots, x_n)$ that relates the value of the desired difficult-to-estimate quantity y with the values of easier-to-estimate auxiliary quantities $x_1, \ldots, x_n$.
- We also have expert knowledge about each of the quantities x_i. For each i, this knowledge is described in terms of the corresponding membership function $\mu_i(x_i)$. For each i and for each value x_i, the value $\mu_i(x_i)$ is the degree of confidence that this value is indeed a possible value of the i-th quantity.

Based on this information, we want to find the membership function $\mu(y)$ which describes, for each real number y, the degree of confidence that this number is a possible value of the desired quantity.

As we have mentioned earlier, y is a possible value of the desired quantity if for some values $x_1, \ldots, x_n$, x_1 is a possible value of the first input quantity, and x_2 is a possible value of the second input quantity, $\ldots$, and $y = f(x_1 \ldots, x_n)$. We know that the degree of confidence that x_1 is a possible value of the first input quantity is equal to $\mu_1(x_1)$, that the degree of confidence that x_2 is a possible value of the second input quantity is equal to $\mu_2(x_2)$, etc. The degree of confidence $d(y, x_1, \ldots, x_n)$ in an equality $y = f(x_1 \ldots, x_n)$ is, of course, equal to 1 if this equality holds, and to 0 if this equality does not hold.

We have already agreed to represent "and" as min. Thus, for each combination of values $x_1, \ldots, x_n$, the degree of confidence in a composite statement "x_1 is a possible value of the first input quantity, and x_2 is a possible value of the second input quantity, $\ldots$, and $y = f(x_1 \ldots, x_n)$" is equal to

$$\min(\mu_1(x_1), \mu_2(x_2), \ldots, d(y, x_1, \ldots, x_n)).$$

We can simplify this expression if we consider two possible cases: when the equality $y = f(x_1 \ldots, x_n)$ holds, and when this equality does not hold.

When the equality $y = f(x_1 \ldots, x_n)$ holds, we get $d(y, x_1, \ldots, x_n) = 1$, and thus, the above degree of confidence is simply equal to

$$\min(\mu_1(x_1), \mu_2(x_2), \ldots, \mu_n(x_n)).$$

When the equality $y = f(x_1 \ldots, x_n)$ does not hold, we get $d(y, x_1, \ldots, x_n) = 0$, and thus, the above degree of confidence is simply equal to 0.

We want to combine these degrees of belief into a single degree of confidence that "for some values $x_1, \ldots, x_n$, x_1 is a possible value of the first input quantity, and x_2 is a possible value of the first input quantity, ..., and $y = f(x_1 \ldots, x_n)$". The words "for some values $x_1, \ldots, x_n$" means that the following composite property holds either for one combination of real numbers $x_1, \ldots, x_n$, or for another combination – until we exhaust all (infinitely many) such combinations. We have already agreed to represent "or" as max. Thus, the desired degree of confidence $\mu(y)$ is equal to the maximum of the degrees corresponding to different combinations $x_1, \ldots, x_n$. Since we have infinitely many possible combinations, the maximum is not necessarily attained, so we should, in general, consider supremum instead of maximum:

$$\mu(y) = \sup \min(\mu_1(x_1), \mu_2(x_2), \ldots, d(y, x_1, \ldots, x_n)),$$

where the supremum is taken over all possible combinations.

Since we know that the maximized degree is non-zero only when $y = f(x_1 \ldots, x_n)$, it is sufficient to only take supremum over such combinations. For such combinations, we can omit the term $d(y, x_1, \ldots, x_n)$ in the maximized expression, so we arrive at the following formula:

$$\mu(y) = \sup\{\min(\mu_1(x_1), \mu_2(x_2), \ldots, \mu_n(x_n)) : y = f(x_1, \ldots, x_n)\}.$$

This formula describes a reasonable way to extend an arbitrary data processing algorithm $f(x_1, \ldots, x_n)$ from real-valued inputs to a more general case of fuzzy inputs. It was first proposed by L. Zadeh and is thus called *Zadeh's extension principle*.

This is the main formula that describes knowledge processing under fuzzy uncertainty. In the following section, we will show that from the computational viewpoint, the application of this formula can be reduced to interval computations – and indeed, this is how knowledge processing under fuzzy uncertainty is usually done, by using this reduction; see, e.g., [1, 3, 7].

4 Main Relation between Interval Computing and Soft Computing: Fuzzy-Related Knowledge Processing Can Be Reduced to Interval Computations

4.1 An Alternative Set Representation of a Membership Function: Alpha-Cuts

To describe the desired relation between fuzzy and interval data processing, we must first reformulate fuzzy techniques in an interval-related form.

In some situations, an expert knows exactly which values of x_i are possible and which are not. In this situation, the expert's knowledge can be naturally represented by describing the *set* of all possible values.

In general, the expert's knowledge is fuzzy:

- we may still have some values about which the expert 100% believes that they are possible, and
- we may still have some values about which the expert 100% believes that they are impossible, but
- in general, the expert is not 100% confident about which values of x_i are possible and which are not.

For example, a geophysicist may be confident that the density x_i of some mineral can take on values ranging from 3.4 to 3.7 g/cm^3, and she may know that values smaller than 3.0 or larger than 4.0 are absolutely impossible, but she is not sure whether values from 3.0 to 3.4 or from 3.7 to 4.0 are indeed realistically possible.

As we have mentioned, the ultimate purpose of the measurements and estimates is to make decisions. In the geophysical example, we have measured the density at a certain depth, and we need to decide:

- whether it is possible that we have the desired mineral – in which case we should undertake more measurements, or
- whether it is not possible that we have the desired mineral – in which case we should not waste our resources on this region and move to more promising regions.

In practice, decisions are made under uncertainty. If we only have a fuzzy expert description of possible values – in terms of the membership function $\mu_S(x_i)$ – which values x_i should we then classify as possible ones and which as impossible?

Under uncertainty, a reasonable idea is to select a threshold $\alpha \in (0, 1]$. In this case,

- all the values x_i for which the expert's degree of confidence is strong enough – i.e., for which $\mu_S(x_i) \geq \alpha$ – are classified as possible;
- similarly, all the values x_i for which the expert's degree of confidence is not sufficiently strong – i.e., for which $\mu_S(x_i) < \alpha$ – are classified as impossible.

The resulting set of possible elements

$$\mathbf{x}_i(\alpha) \stackrel{\text{def}}{=} \{x_i : \mu_S(x_i) \geq \alpha\}$$

is called the α-*cut* of the membership function $\mu_S(x_i)$.

The choice of a threshold α depends on the practical problem. For example, if we are looking for a potentially very valuable mineral deposit, then it makes sense to continue prospecting even when our degree of confidence is not very high. In this case, it makes sense to select a reasonably small threshold α.

On the other hand, if the potential benefit is not high and our resources are limited, it makes sense to limit our search to highly promising regions – i.e., to select a reasonably high threshold α.

To adequately describe the expert knowledge irrespective of an application, we therefore need to know the α-cuts corresponding to different thresholds α. Each α-cut $\mathbf{x}_i(\alpha)$ describes the set of values which are possible with degree of confidence at least α.

By definition, α-cuts corresponding to different α are *nested*: when $\alpha \le \alpha'$, then $\mu_S(x_i) \ge \alpha'$ implies $\mu_S(x_i) \ge \alpha$ and thus,

$$\mathbf{x}_i(\alpha') = \{x_i : \mu_S(x_i) \ge \alpha'\} \subseteq \mathbf{x}_i(\alpha) = \{x_i : \mu_S(x_i) \ge \alpha\}.$$

Comment. It is worth mentioning that if we know the α-cuts

$$\mathbf{x}_i(\alpha) = \{x_i : \mu_S(x_i) \ge \alpha\}$$

corresponding to all possible values $\alpha \in (0, 1]$, then we can uniquely reconstruct the corresponding membership function $\mu_S(x_i)$. The possibility for such a reconstruction follows from the fact that every real number r is equal to the largest largest value α for which $r \ge \alpha$. In particular, for every x_i, the value $\mu_S(x_i)$ is equal to the largest value α for which $\mu_S(x_i) \ge \alpha$. By definition of the α-cut, the inequality $\mu_S(x_i) \ge \alpha$ is equivalent to $x_i \in \mathbf{x}_i(\alpha)$. Thus, for every x_i, the value $\mu_S(x_i)$ can be reconstructed as the largest value α for which $x_i \in \mathbf{x}_i(\alpha)$.

Thus, we can alternatively view a membership function as a nested family of α-cuts; see, e.g., [3].

4.2 Fuzzy Numbers and Intervals

In most practical situations, the membership function starts with 0, continuously increases until a certain value and then continuously decreases to 0. Such membership function describe usual expert's expressions such as "small", "medium", "reasonably high", "approximately equal to a with an error about σ", etc. Such examples were given in the previous chapter. Since membership functions of this type are actively used in expert estimates of number-valued quantities, they are usually called *fuzzy numbers*.

For a fuzzy number $\mu_i(x_i)$, every α-cut $\mathbf{x}_i(\alpha)$ is an interval. Thus, a fuzzy number can be viewed as a nested family of intervals $\mathbf{x}_i(\alpha)$ corresponding to different degrees of confidence.

4.3 Simplest "And"- and "Or"-Operations: Reformulation in Terms of Sets and Alpha-Cuts

The main formulas for fuzzy computations (i.e., for processing fuzzy data) were derived by using the simplest "and"- and "or"-operations $f_{\&}(d_1, d_2) =$

$\min(d_1, d_2)$ and $f_\vee(d_1, d_2) = \max(d_1, d_2)$. Thus, before we describe how fuzzy computations can be reduced to interval computations, let us first reformulate these "and"- and "or"-operations in terms of α-cuts.

Specifically, let us assume that we have two properties A and B which are described by the membership functions $\mu_A(x)$ and $\mu_B(x)$ and, correspondingly, by the α-cuts $\mathbf{x}_A(\alpha) = \{x : \mu_A(x) \geq \alpha\}$ and $\mathbf{x}_B(\alpha) = \{x : \mu_B(x) \geq \alpha\}$. If we use the simplest "and"-operation $f_\&(d_1, d_2) = \min(d_1, d_2)$, then the composite property $A \,\&\, B$ ("A and B") is described by the membership function $\mu_{A\,\&\,B}(x) = \min(\mu_A(x), \mu_B(x))$. What are the α-cuts

$$\mathbf{x}_{A\,\&\,B}(\alpha) = \{x : \mu_{A\,\&\,B}(x) \geq \alpha\}$$

corresponding to this membership function?

The minimum of two real numbers is greater than or equal to α if and only if both of these numbers are greater than or equal to α. Thus, the condition $\mu_{A\,\&\,B}(x) = \min(\mu_A(x), \mu_B(x)) \geq \alpha$ is equivalent to "$\mu_A(x) \geq \alpha$ and $\mu_B(x) \geq \alpha$". Hence, the set $\mathbf{x}_{A\,\&\,B}(\alpha)$ of all the values x for which the condition $\mu_{A\,\&\,B}(x) = \min(\mu_A(x), \mu_B(x)) \geq \alpha$ is satisfied can be found simply as the intersection of the set of all x for which $\mu_A(x) \geq \alpha$ and the set of all x for which $\mu_B(x) \geq \alpha$. In other words, for every α, we have

$$\mathbf{x}_{A\,\&\,B}(\alpha) = \mathbf{x}_A(\alpha) \cap \mathbf{x}_B(\alpha).$$

Therefore, to perform the simplest "and"-operation $f_\&(d_1, d_2) = \min(d_1, d_2)$, we simply take the intersection of the corresponding α-cuts. This is a very natural operation, since, for exactly defined sets and properties, the set of all the elements which satisfy the property $A \,\&\, B$ is equal to the intersection of the set of all elements which satisfy property A and the set of all elements which satisfy property B.

Similarly, for the simplest "or"-operation $f_\vee(d_1, d_2) = \max(d_1, d_2)$, the composite property $A \vee B$ ("A or B") is described by the membership function $\mu_{A\vee B}(x) = \max(\mu_A(x), \mu_B(x))$. To find the α-cuts

$$\mathbf{x}_{A\vee B}(\alpha) = \{x : \mu_{A\vee B}(x) \geq \alpha\}$$

corresponding to this membership function, we can use the fact that the maximum of two real numbers is greater than or equal to α if and only if one of these numbers is greater than or equal to α. Thus, the condition $\mu_{A\vee B}(x) = \max(\mu_A(x), \mu_B(x)) \geq \alpha$ is equivalent to "$\mu_A(x) \geq \alpha$ or $\mu_B(x) \geq \alpha$". Hence, the set $\mathbf{x}_{A\vee B}(\alpha)$ of all the values x for which the condition $\mu_{A\vee B}(x) = \max(\mu_A(x), \mu_B(x)) \geq \alpha$ is satisfied can be found simply as the union of the set of all x for which $\mu_A(x) \geq \alpha$ and the set of all x for which $\mu_B(x) \geq \alpha$. In other words, for every α, we have

$$\mathbf{x}_{A\vee B}(\alpha) = \mathbf{x}_A(\alpha) \cup \mathbf{x}_B(\alpha).$$

Therefore, to perform the simplest "or"-operation $f_\vee(d_1, d_2) = \max(d_1, d_2)$, we simply take the union of the corresponding α-cuts. This is also a very

natural operation, since, for exactly defined sets and properties, the set of all the elements which satisfy the property $A \vee B$ is equal to the union of the set of all elements which satisfy property A and the set of all elements which satisfy property B.

4.4 Fuzzy Computations Can Be Reduced to Interval Computations: Derivation

The main problem of fuzzy computation can be described as follows:

- We know an algorithm $y = f(x_1, \ldots, x_n)$ that relates the value of the desired difficult-to-estimate quantity y with the values of easier-to-estimate auxiliary quantities $x_1, \ldots, x_n$.
- We also know, for every i from 1 to n, a membership function $\mu_i(x_i)$ which describes the expert knowledge about the i-th input quantity x_i.

Our objective is to compute the function

$$\mu(y) = \sup\{\min(\mu_1(x_1), \mu_2(x_2), \ldots, \mu_n(x_n)) : y = f(x_1, \ldots, x_n)\}.$$

Let us now describe this relation in terms of α-cuts. This description will constitute the main relation between fuzzy and interval computing. This relation was first discovered and proved in [2]. To describe this result in precise terms, let us first make some mathematics-related remarks.

The function $y = f(x_1, \ldots, x_n)$ describes the relation between physical quantities. In physics, such a relation is usually continuous. Even when we have seemingly discontinuous transitions, e.g., in phase transitions when, say, the density of water changes into a much smaller density of steam, it is not really a discontinuous transition, it is simply a very fast but still continuous one. In view of this observation, we will assume that the function $y = f(x_1, \ldots, x_n)$ is continuous.

We will also assume the membership functions $\mu_i(x_i)$ are continuous. If we had exact knowledge, then continuity would make no sense, since then the corresponding degree of confidence would abruptly go from 1 for possible values to 0 for impossible ones, without ever attaining any intermediate degrees. However, for fuzzy knowledge, continuity makes perfect sense. If there is some degree of confidence that a value x_i is possible, then it makes sense to assume that values close to x_i are possible too – with a similar degree of belief. In practice, as we mentioned earlier in this chapter and in the previous chapter, membership functions are indeed usually continuous.

It is important to mention that for continuous membership functions $\mu_i(x_i)$, α-cuts $\{x_i : \mu_i(x_i) \geq \alpha\}$ are closed sets (i.e., sets which contain all their limit points).

Finally, we require that for every i and for every $\alpha > 0$, the α-cut is a compact set. For real numbers, since we have already assumed that the α-cuts $\{x_i : \mu_i(x_i) \geq \alpha\}$ are closed sets, it is sufficient to require that these sets

are bounded. This is true, e.g., if we assume that all the membership functions correspond to fuzzy numbers; in this case, all α-cuts are intervals.

Suppose that we know the α-cuts $\mathbf{x}_i(\alpha)$ corresponding to the inputs, and we want to find the α-cuts $\mathbf{y}(\alpha)$ corresponding to the output. By definition of an α-cut, $y \in \mathbf{y}(\alpha)$ means that $\mu(y) \geq \alpha$, i.e., that

$$\sup\{\min(\mu_1(x_1), \mu_2(x_2), \ldots, \mu_n(x_n)) : y = f(x_1, \ldots, x_n)\} \geq \alpha.$$

By definition of the supremum, this means that for every integer $k > 2/\alpha$, there exists a tuple $(x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)})$ for which $y = f(x_1^{(k)}, \ldots, x_n^{(k)})$ and

$$\min(\mu_1(x_1^{(k)}), \mu_2(x_2^{(k)}), \ldots) \geq \alpha - 1/k.$$

The minimum of several numbers is $\geq \alpha - 1/k$ if and only if all these numbers are $\geq \alpha - 1/k$, i.e., $\mu_i(x_i^{(k)}) \geq \alpha - 1/k$ for all i. Since $k > 2/\alpha$, we have $1/k < \alpha/2$ and $\alpha - 1/k > \alpha/2$. Thus, for each i and all k, the value $x_i^{(k)}$ belongs to the compact $(\alpha/2)$-cut $\mathbf{x}_i(\alpha/2)$. Since the tuples $(x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)})$ belong to the compact set

$$\mathbf{x}_1(\alpha/2) \times \mathbf{x}_2(\alpha/2) \times \ldots \times \mathbf{x}_n(\alpha/2),$$

the sequence of these tuples has a convergent subsequence converging to some tuple $(x_1, x_2, \ldots, x_n)$. Since both f and μ_i are continuous, for this limit tuple, we get $y = f(x_1, \ldots, x_n)$ and $\mu_i(x_i) \geq \alpha$. In other words, every element $y \in \mathbf{y}(\alpha)$ can be represented as $y = f(x_1, \ldots, x_n)$ for some values $x_i \in \mathbf{x}_i(\alpha)$.

Conversely, if $x_i \in \mathbf{x}_i(\alpha)$ and $y = f(x_1, \ldots, x_n)$, then $\mu_i(x_i) \geq \alpha$ and therefore, $\min(\mu_1(x_1), \mu_2(x_2), \ldots, \mu_n(x_n)) \geq \alpha$ and hence

$$\sup\{\min(\mu_1(x_1), \mu_2(x_2), \ldots, \mu_n(x_n)) : y = f(x_1, \ldots, x_n)\} \geq \alpha,$$

i.e., $\mu(y) \geq \alpha$ and $y \in \mathbf{y}(\alpha)$

Thus, the desired α-cut $\mathbf{y}(\alpha)$ consists of exactly values $y = f(x_1, \ldots, x_n)$ for $x_i \in \mathbf{x}_i(\alpha)$:

$$\mathbf{y}(\alpha) = \{f(x_1, \ldots, x_n) : x_1 \in \mathbf{x}_1(\alpha), \ldots, x_n \in \mathbf{x}_n(\alpha)\}.$$

This is exactly the range that we defined when we described interval computations, so we can rewrite this formula as

$$\mathbf{y}(\alpha) = f(\mathbf{x}_1(\alpha), \ldots, \mathbf{x}_n(\alpha)).$$

In particular, for fuzzy numbers, when all α-cuts $\mathbf{x}_i(\alpha)$ are intervals, computing each α-cut $\mathbf{y}(\alpha)$ is exactly the problem of interval computations.

4.5 Fuzzy Computations Can Be Reduced to Interval Computations: Conclusion

If the inputs $\mu_i(x_i)$ are fuzzy numbers and the function $y = f(x_1, \ldots, x_n)$ is continuous, then for each α, the α-cut $\mathbf{y}(\alpha)$ of y is equal to the range of possible values of $f(x_1, \ldots, x_n)$ as x_i ranges over $\mathbf{x}_i(\alpha)$ for all i:

$$\mathbf{y}(\alpha) = f(\mathbf{x}_1(\alpha), \ldots, \mathbf{x}_n(\alpha)).$$

Thus, from the computational point of view, the problem of processing data under fuzzy uncertainty can be reduced to several problems of data processing under interval uncertainty – as many problems as there are α-levels. As we have mentioned, this is not just a theoretical observation: this is exactly how fuzzy data processing is usually performed, and this is how interval computations techniques are explained in fuzzy textbooks.

5 Auxiliary Relation between Interval Computing and Soft Computing: Interval-Valued Fuzzy Techniques

5.1 Intervals are Necessary to Describe Degrees of Belief

Above, we described an idealized situation, in which we can describe degrees of belief by exact real numbers. In practice, the situation is more complicated, because experts cannot describe their degrees of belief precisely; see, e.g., [6] and references therein.

Indeed, let us start by reviewing the above-described methods of eliciting degrees of belief. If an expert describes his or her degree of belief by selecting, e.g., 8 on a scale from 0 to 10, this does not mean that his or her degree of belief is exactly 0.8: if instead, we ask him or her to select on a scale from 0 to 9, then whatever he or she chooses, after dividing it by 9, we will never get 0.8. If an expert chooses a value 8 on a 0 to 10 scale, then the only thing that we know about the expert's degree of belief is that it is closer to 8 than to 7 or to 9, i.e., that this degree of belief belongs to the *interval* $[0.75, 0.85]$.

Another possible source of interval uncertainty is when we have *several* experts, and their estimates differ. If, e.g., two equally good experts point to 7 and 8, then, if we are cautious, we would rather describe the resulting degree of belief as the interval $[0.7, 0.8]$ (or, in view of the above remark, as the interval $[0,65, 0.85]$).

If we determine the degree of belief by polling, then the same argument shows that the resulting numbers are not precise: e.g., if 8 out of 10 experts voted for A, then we cannot say that the actual degree of belief is exactly 0.8, because, if we repeated this procedure with 9 experts, we will never get exactly 0.8. In this case, there are two other sources of uncertainty: First, picking experts is sort of a random procedure, so, the result of voting is a statistical estimate that is not precise (just like a statistical frequency estimate

of probability). A better description will be to give an *interval* of possible values of $d(A)$.

The polling method of estimating the degree of belief is based on the assumption that an expert can always tell whether he believes in a given statement S or not. Then, we take the ratio $d(S) = N(S)/N$ of the number $N(S)$ of experts who believe in S to the total number N of experts as the desired estimate. For $\neg S$, we thus have $N(\neg S) = N - N(S)$, so $d(\neg S) = N(\neg S)/N = 1 - d(S)$. In reality, an expert is often unsure about S. In this case, instead of dividing the experts into two categories: those who believe in S and those who do not, we must divide them into *three* categories: those who believe that S is true (we will denote their number by $N(S)$), those who believe that S is false (we will denote their number by $N(\neg S)$), and those who do not have the definite opinion about S; there are $N - N(S) - N(\neg S)$ of them. In this situation, one number is not sufficient to describe the experts' degree of belief in S, we need at least two. There are two ways to describe it: We can describe the degree of belief in S as $d(S) = N(S)/N$ and the degree of belief in $\neg S$ as $d(\neg S) = N(\neg S)/N$. These two numbers must satisfy the condition $d(S) + d(\neg S) \le 1$. This description is known as *intuitionistic fuzzy logic*. (The reason for the word "intuitionistic" is that this logic is close to the original intuitionistic idea that the law of excluded middle is not always true.)

Alternatively, we can describe the degree of belief $d(S)$ in S and the degree of *plausibility* of S estimating as the fraction of experts who do not consider S impossible, i.e., as $pl(S) = 1 - d(\neg S)$, i.e., as an interval $[d(S), pl(S)]$. This representation corresponds to the *Dempster-Shafer formalism* (see Chapter 2).

So, to describe degrees of belief adequately, we must use *intervals* instead of real numbers.

5.2 Interval Computations for Processing Interval-Valued Degrees of Belief: General Idea

For an expert system with interval-valued degrees of belief, the following problem arises: suppose that we have an expert system whose knowledge base consists of statements $S_1, \ldots, S_N$, and we have an algorithm $f(Q, d_1, \ldots, d_N)$ (called *inference engine*) that for any given query Q, transforms the degrees of belief $d(S_1), \ldots, d(S_N)$ in the statements from the knowledge base into a degree of belief $d(Q) = f(Q, d(S_1), \ldots, d(S_N))$ in Q (for example, if $Q = S_1 \& S_2$, then $f(d_1, \ldots, d_N) = f_{\&}(d_1, d_2)$). Suppose now that we know only the *intervals* $\mathbf{d}(S_1)$, $\ldots$, $\mathbf{d}(S_N)$ that contain the desired degree of belief. Then, the degree of belief in Q can take any value from the set

$$f(Q, \mathbf{d}(S_1), \ldots, \mathbf{d}(S_N)) = \{f(Q, d_1, \ldots, d_N) \,|\, d_i \in \mathbf{d}(S_i)\}.$$

Computing such an interval is a typical problem of *interval computations*.

In particular, since the functions $f_{\&}$ and $f_{\vee}$ are increasing in both arguments, we have

$$f_{\&}([\underline{x},\overline{x}],[\underline{y},\overline{y}]) = [f_{\&}(\underline{x},\underline{y}), f_{\&}(\overline{x},\overline{y})]$$

and

$$f_{\vee}([\underline{x},\overline{x}],[\underline{y},\overline{y}]) = [f_{\vee}(\underline{x},\underline{y}), f_{\vee}(\overline{x},\overline{y})].$$

For example,

$$\min([\underline{x},\overline{x}],[\underline{y},\overline{y}]) = [\min(\underline{x},\underline{y}), \min(\overline{x},\overline{y})]$$

and

$$\max([\underline{x},\overline{x}],[\underline{y},\overline{y}]) = [\max(\underline{x},\underline{y}), \max(\overline{x},\overline{y})].$$

In the following chapters, we will give examples of practical applications of interval-valued fuzzy values.

6 Conclusion

In this chapter, we have explained intrinsic and useful relations between interval computing and soft computing – specifically, fuzzy data processing. The main relation is that a fuzzy set (membership function) can be viewed as a nested family of intervals – its α-cuts corresponding to different levels of uncertainty α. From the computational viewpoint, fuzzy data processing can be (and usually is) reduced to level-by-level interval computations with the corresponding α-cuts.

Another relation comes from the fact that it is usually difficult to describe experts' degrees of certainty by exact real numbers. A more adequate description of expert's uncertainty is by an interval. Processing interval-valued degrees of uncertainty also requires interval computations.

References

1. Klir, G.J., Yuan, B.: Fuzzy Sets and Fuzzy Logic. Prentice Hall (1995)
2. Nguyen, H.T.: A note on the extension principle for fuzzy sets. J. Math. Anal. and Appl. **64**, 369–380 (1978)
3. Nguyen, H.T., Kreinovich, V.: Nested intervals and sets: concepts, relations to fuzzy sets, and applications. In: R.B. Kearfott, V. Kreinovich (eds.) Applications Of Interval Computations, pp. 245–290. Kluwer Academic Publishers Group, Norwell, MA, USA, and Dordrecht, The Netherlands (1996)
4. Nguyen, H.T., Kreinovich, V.: Applications Of Continuous Mathematics To Computer Science. Kluwer Academic Publishers Group, Norwell, MA, USA, and Dordrecht, The Netherlands (1997)
5. Nguyen, H.T., Kreinovich, V.: Methodology of fuzzy control: an introduction. In: H.T. Nguyen, M. Sugeno (eds.) Fuzzy Systems: Modeling and Control, pp. 19–62. Kluwer Academic Publishers Group, Norwell, MA, USA, and Dordrecht, The Netherlands (1998)

6. Nguyen, H.T., Kreinovich, V., Zuo, Q.: Interval-valued degrees of belief: applications of interval computations to expert systems and intelligent control. International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems (IJUFKS) **5**, 317–358 (1997)
7. Nguyen, H.T., Walker, E.A.: First Course In Fuzzy Logic. CRC Press, 2000 N.W. Corporate Blvd., Boca Raton, FL 33431-9868, USA (2006)
8. Rabinovich, S.: Measurement Errors and Uncertainties: Theory and Practice. American Institute of Physics, New York, NY, USA (2005)