Towards Efficient Algorithms for Approximating a Fuzzy Relation by Fuzzy Rules: Case When "And"- and "Or"-Operation are Distributive

Christian Servin[1,2]
[1]Information Technology Dept.
El Paso Community College
919 Hunter, El Paso, Texas 79915, USA
Email: cservin@gmail.com

Vladik Kreinovich[2]
[2]TRACS Center, Dept. of Computer Science
University of Texas at El Paso
El Paso, Texas 79968, USA
Email: vladik@utep.edu

***Abstract*—A generic fuzzy relation often requires too many parameters to represent – especially when we have a relation between many different quantities $x_1, \ldots, x_n$. There is, however, a class of relations which require much fewer parameters to describe - namely, relations which come from fuzzy rules. It is therefore reasonable to approximate a given relation by fuzzy rules. In this paper, we explain how this can be done in an important case when "and"- and "or"-operation are distributive – and we also explain why this case is important.**

I. Formulation of the Problem

Relations are ubiquitous. Many real-life quantities $x_1, \ldots, x_n$ are related – in the sense that once we know the value of one or more of the quantities, this knowledge restricts possible values of other quantities.

In some cases, we have a *functional* relation – when the values of the quantities $x_1, \ldots, x_{n-1}$ uniquely determine the value of the quantity x_n. For example, according to Ohm's law, the voltage V is uniquely determine by the current I and the resistance R as $V = I \cdot R$.

In many other cases, however, we have relations which are not functional. In other words, even if we know the exact values of all the quantities $x_1, \ldots, x_{n-1}$, we can still have different possible values of x_n. This is actually true even for voltage: different materials exhibit minor deviation from the Ohm's law; as a result, even if we know the current and the resistance, we can only conclude that the voltage V is close to $I \cdot R$ (e.g., that V can only takes values from the interval $[I \cdot R - \varepsilon, I \cdot R + \varepsilon]$ for some small $\varepsilon > 0$.

In mathematical terms, a relation between real-valued quantities x_i is usually defined as a mapping $R : \mathbb{R}^n \to \{0, 1\}$ such that $R(x_1, \ldots, x_n) = 1$ indicates that the corresponding combination of values $(x_1, \ldots, x_n)$ is possible in a real-life situation.

Real-life relations are often fuzzy. In practice, about some combinations $(x_1, \ldots, x_n)$, we are often not 100% sure whether these combinations are possible or not. In the traditional ("crisp") approach, we simply count all such combinations as possible – since there is a possibility that such combinations occur. However, this crisp representation ignores the fact that we may be more certain about the possibility of some combinations and less certain about the possibility of others. To describe the different, it is necessary to know, for each possible combination $(x_1, \ldots, x_n)$, our degree of certainty that this combination is practically possible.

In the computer, "true" is usually represented as 1, and "false" as 0. It is therefore natural to represent intermediate degrees of certainty as numbers from the interval $[0, 1]$: the larger the number, the larger our degree of confidence. The resulting mapping $R : \mathbb{R}^n \to \{0, 1\}$ is known as a *fuzzy relation*; see, e.g., [3], [4], [6].

Need for a concise representation of fuzzy relation. To use the information about the relation, we need to represent it in a computer. Theoretically, each of the quantities x_i can have infinitely many different values, but in practice, due to inevitable measurement uncertainty, for each variable x_i, we can only have finitely many distinguishable values $x_{i1}, \ldots, x_{ij}, \ldots, x_{iN_i}$. Because of this, knowing a relation means that we know the values $R(x_{1i_1}, \ldots, x_{ni_n})$ corresponding to all possible combinations $(x_{1i_1}, \ldots, x_{ni_n})$.

In principle, we can simply store the degrees of certainty corresponding to all possible $N_1 \cdot \ldots \cdot N_n$ combinations. This requires storing and processing $\approx N^n$ values, where N is a typical number of distinct values of each quantity. The problem with this representation is that, as we have mentioned, many quantities are related to each other; so, to have the most adequate representation of a real-life phenomenon, we need to describe a relation between a large number of variables. When n is large, the resulting number of values N^n grows exponentially with n – and, as it is well known about exponential functions, the numbers easily become astronomically high, exceeding the ability of modern computers to store and/or process this information; see, e.g., [5]. We therefore need to come up with a more concise representation of fuzzy relations.

An approximate representation is OK. The degrees of certainty can only be approximately described: an expert cannot realistically distinguish between his/her degree of 0.71 and 0.72 :-) Since the values are only approximately known anyway, it is OK to represent them approximately. This possibility of using an approximate representation provides the flexibility which makes more concise representations possible.

Fuzzy rules as a natural concise representation of fuzzy relations. Many fuzzy relations come from *fuzzy rules*, i.e., from a combination of rules of the type

$$\text{“if } A_{r,1}(x_1) \text{ and } \ldots \text{and } A_{r,n-1}(x_{n-1}) \text{ then } A_{r,n}(x_n)\text{”},$$

where $r = 1, \ldots, n_r$ is the number of the rule, and $A_{r,i}(x_i)$ are fuzzy properties. The ubiquity of such rules comes from the fact that this is how experts often describe their decisions. For example, a driver can explain his or her driving strategy by describing rules like “if a car in front is close, and it starts breaking seriously, one needs to hit the brakes hard right away”. Such rules use imprecise (fuzzy) words like “close”, “seriously”, “hard”, which are naturally described by fuzzy logic techniques.

One of the most common ways to formalize the fuzzy rules it the *Mamdani approach*. In this approach, we take into account that a tuple $(x_1, \ldots, x_n)$ is consistent with the rules if for one of the given rules, the conditions are satisfied and the conclusion is satisfied as well. In other words, a tuple $(x_1, \ldots, x_n)$ is consistent with the given rules if and only if the following statement is true:

$$(A_{1,1}(x_1) \,\&\, \ldots \,\&\, A_{1,n-1}(x_{n-1}) \,\&\, A_{1,n}(x_n)) \vee \ldots \vee$$
$$(A_{n_r,1}(x_1) \,\&\, \ldots \,\&\, A_{n_r,n-1}(x_{n-1}) \,\&\, A_{n_r,n}(x_n)).$$

Fuzzy logic techniques enable us to transform this formula into the exact value of a degree $d(x_1, \ldots, x_n)$ to which the tuple $(x_1, \ldots, x_n)$ is consistent with the rules. Specifically:

- we can use an “and”-operation (t-norm) $f_\&(a, b)$ to represent “and”, and
- we can use an “or”-operation (t-conorm) $f_\vee(a, b)$ to represent “or”.

As a result, we get the following degree:

$$d(x_1, \ldots, x_n) = f_\vee(d_1(x_1, \ldots, x_n), \ldots, d_{n_r}(x_1, \ldots, x_n)),$$

where the degree $d_r(x_1, \ldots, x_n)$ to which the tuple $(x_1, \ldots, x_n)$ is consistent with the r-th rule is equal to

$$d_r(x_1, \ldots, x_n) =$$
$$f_\&(A_{r,1}(x_1), \ldots, A_{r,n-1}(x_{n-1}), A_{r,n}(x_n)).$$

Fuzzy rules are a natural concise way of representing a relation. Thus, it is reasonable to try to approximate a given fuzzy relation by an appropriate family of rules.

What we do in this paper. In this paper, we propose new algorithms for representing a given fuzzy relation in terms of fuzzy rules, algorithms which are applicable in the important case when “and”- and “or”-operations are distributive.

II. Why Is Distributivity Important?

Before we start describing our algorithms, we need to explain why the case when “and”- and “or”-operations are distributive is important. To explain this importance, let us first recall the motivations behind the usual definitions of “and”-operations (t-norms) and “or”-operations (t-conorms).

Why t-norms and t-conorms: reminder. The main idea behind “and”-operations is that often, we know the expert’s degrees of confidence $a = d(A)$ and $b = d(B)$ in two statements A and B, and we want to estimate the expert’s degree of confidence in a composite statement $A \,\&\, B$ or $A \vee B$. The only information that we have for this estimate consist of degrees a and b, so the resulting estimates are obtained by applying some computations to these two numbers:

- the algorithm for producing the estimate for $d(A \,\&\, B)$ is denoted by $f_\&(a, b)$, so the desired estimate has the form $f_\&(d(A), d(B))$; and
- the algorithm for producing the estimate for $d(A \vee B)$ is denoted by $f_\vee(a, b)$, so the desired estimate has the form $f_\vee(d(A), d(B))$.

Which properties should the corresponding functions $f_\&(a, b)$ and $f_\vee(a, b)$ satisfy?

The composite statements $A \,\&\, B$ and $B \,\&\, A$ are equivalent to each other for every two statements A and B. It is therefore reasonable to require that the estimates $f_\&(a, b)$ and $f_\&(b, a)$ for the expert’s degree of belief in these composite statements coincide, i.e., that $f_\&(a, b) = f_\&(b, a)$. In mathematical terms, this means that the “and”-operation $f_\&(a, b)$ should be *commutative*.

Similarly, since for every two statements A and B, the composite statements $A \vee B$ and $B \vee A$ are equivalent to each other, it is reasonable to require that the estimates $f_\vee(a, b)$ and $f_\vee(b, a)$ for the expert’s degree of belief in these composite statements coincide, i.e., that $f_\vee(a, b) = f_\vee(b, a)$. Thus, the “or”-operation $f_\vee(a, b)$ should also be commutative.

Another pairs of equivalent statements are $(A \,\&\, B) \,\&\, C$ and $A \,\&\, (B \,\&\, C)$. We can estimate the expert’s degree of belief in the statement $(A \,\&\, B) \,\&\, C$ if:

- first, we apply the “and”-operation to the degrees $a = d(A)$ and $b = d(B)$ and get an estimate $f_\&(a, b)$ for the expert’s degree of belief in a statement $A \,\&\, B$;
- then, we apply the same “and”-operation to another pair of numbers:
  - our estimate $f_\&(a, b)$ of the expert’s degree of belief in $A \,\&\, B$, and
  - the expert’s degree of belief $c = d(C)$ in the statement C.

As a result, we get the estimate $f_\&(f_\&(a, b), c)$ for the expert’s degree of belief in $(A \,\&\, B) \,\&\, C$.

Similarly, we can estimate the expert’s degree of belief in the statement $A \,\&\, (B \,\&\, C)$ if:

- first, we apply the “and”-operation to the degrees $b = d(B)$ and $c = d(C)$ and get an estimate $f_\&(b, c)$ for the expert’s degree of belief in a statement $B \,\&\, C$;
- then, we apply the same “and”-operation to another pair of numbers:
  - the expert’s degree of belief $a = d(A)$ in the statement A; and
  - our estimate $f_\&(b, c)$ of the expert’s degree of belief in $B \,\&\, C$.

As a result, we get the estimate $f_\&(a, f_\&(b, c))$ for the expert's degree of belief in $A \,\&\, (B) \,\&\, C)$.

Since the statements $(A \,\&\, B) \,\&\, C$ and $A \,\&\, (B \,\&\, C)$ are equivalent $A \,\&\, (B \vee C)$ and $(A \,\&\, B) \vee (A \,\&\, C)$, it is reasonable to require that the corresponding estimates $f_\&(f_\&(a, b), c)$ and $f_\&(a, f_\&(b, c))$ for the expert's degrees of belief in these statements be equal, i.e., that $f_\&(f_\&(a, b), c) = f_\&(a, f_\&(b, c))$ for all a, b, and c. In mathematical terms, this means that the "and"-operation $f_\&(a, b)$ should be *associative*.

Similarly, since the composite statements $(A \vee B) \vee C$ and $A \vee (B \vee C)$ are equivalent to each other, it makes sense to require that the corresponding estimates $f_\vee(f_\vee(a, b), c)$ and $f_\vee(a, f_\vee(b, c))$ for the expert's degrees of belief in these statements be equal, i.e., the "or"-operation $f_\vee(a, b)$ should also be associative.

Because of associativity, we can simply write $f_\&(a, b, \ldots, c)$ and $f_\vee(a, b, \ldots, c)$ without worrying about the order of the corresponding "and"- and "or"-operations.

Other properties of "and"- and "or"-operations also follow from common sense. For example, from the fact that "true" $\&\, A$ is equivalent to A, we conclude that $f_\&(a, 1) = a$. From the fact that "true" $\vee A$ is equivalent to "true", we conclude that $f_\vee(a, 1) = 1$.

Similarly, from the fact that "false" $\&\, A$ is equivalent to "false", we conclude that $f_\&(a, 0) = a$, and from the fact that "false" $\vee A$ is equivalent to A, we conclude that $f_\vee(a, 0) = a$.

Another example: if an expert increases his/her belief in one or both of the statements A and B, then it is reasonable to assume that the expert's degree of belief in a composite statement $A \,\&\, B$ will either increase or stay the same, but it cannot decrease. In other words, if $a \le a'$ and $b \le b'$, then we should have $f_\&(a, b) \le f_\&(a', b')$. In mathematical terms, this means that the "and"-operation should be a (non-strictly) increasing function of each of its variables.

Similarly, it is reasonable to require that the "or"-operation $f_\vee(a, b)$ is a non-strictly increasing function of each of its variables.

It is also reasonable to require that small changes in degree $a = d(A)$ and $b = d(B)$ should lead to small changes in $d(A \,\&\, B)$. In other words, it is reasonable to require that the "or"-operation be continuous.

Why distributivity is a reasonable requirement. In the previous subsection, we consider equivalences which use only one of the two connectives: either "and" or "or". In logic, there are also equivalences which combine both "and" and "or". One of these properties is *distributivity*. Specifically, for every three statements A, B, and C, the composite statements $A \,\&\, (B \vee C)$ and $(A \,\&\, B) \vee (A \,\&\, C)$ are equivalent to each other. We can estimate the expert's degree of belief in the statement $A \,\&\, (B \vee C)$ if:

- first, we apply the "or"-operation to the degrees $b = d(B)$ and $c = d(C)$ and get an estimate $f_\vee(b, c)$ for the expert's degree of belief in a statement $B \vee C$;
- then, we apply the "and"-operation to the following pair of numbers:
  - the expert's degree of belief $a = d(A)$ in the statement A, and
  - our estimate $f_\vee(b, c)$ of the expert's degree of belief in $B \vee C$.

As a result, we get the estimate $f_\&(a, f_\vee(b, c))$ for the expert's degree of belief in $A \,\&\, (B \vee C)$.

Similarly, we can estimate the expert's degree of belief in the statement $(A \,\&\, B) \vee (A \,\&\, C)$ if:

- first, we apply the "and"-operation to the degrees $a = d(A)$ and $b = d(B)$ and get an estimate $f_\&(a, b)$ for the expert's degree of belief in a statement $A \,\&\, B$;
- then, we apply the same "and"-operation to the degrees $a = d(A)$ and $c = d(C)$ and get an estimate $f_\&(b, c)$ for the expert's degree of belief in a statement $B \,\&\, C$;
- finally, we apply the "or"-operation to the following pair of numbers:
  - our estimate $f_\&(a, b)$ of the expert's degree of belief in $A \,\&\, B$, and
  - our estimate $f_\&(a, c)$ of the expert's degree of belief in $A \,\&\, C$.

As a result, we get the estimate $f_\vee(f_\&(a, b), f_\&(a, c))$ for the expert's degree of belief in $(A \,\&\, B) \vee (A \,\&\, C)$.

Since the statements $A \,\&\, (B \vee C)$ and $(A \,\&\, B) \vee (A \,\&\, C)$ are equivalent to each other, it is reasonable to require that the corresponding estimates $f_\&(a, f_\vee(b, c))$ and $f_\vee(f_\&(a, b), f_\&(a, c))$ for the expert's degrees of belief in these statements be equal, i.e., that

$$f_\&(a, f_\vee(b, c)) = f_\vee(f_\&(a, b), f_\&(a, c))$$

for all a, b, and c. In mathematical terms, this means that the "and"-operation $f_\&(a, b)$ should be *distributive* over the "and"-operation $f_\vee(a, b)$.

An example of distributive pairs of "and"- and "or"-operations. Let us show that some reasonable pairs of "and"- and "or"-operations do have the distributivity property.

The example is when we use $f_\vee(a, b) = \max(a, b)$ – one of the most frequently used "or"-operations – and an arbitrary "and"-operation $f_\&(a, b)$. Let us show that in this case, we have distributivity, i.e., that $f_\&(a, \max(b, c)) = \max(f_\&(a, b), f_\&(a, c))$ for all a, b, and c.

Indeed, without losing generality, we can assume that $b \le c$ (when $c \le b$, distributivity can be proven in the exact same way). In this case, $\max(b, c) = c$, so the left-hand side of the desired equality is equal to $f_\&(a, c)$:

$$f_\&(a, \max(b, c)) = f_\&(a, c).$$

Due to the fact that the "and"-operation is increasing, $b \le c$ implies that $f_\&(a, b) \le f_\&(a, c)$. Thus, the right-hand side of the desired equality is also equal to $f_\&(a, c)$:

$$\max(f_\&(a, b), f_\&(a, c)) = f_\&(a, c).$$

So, both sides of the desired equality are equal to the same value and are, thus, equal to each other.

Need to restrict ourselves to partial distributivity. The above example looks great, but it turns out that this is the only such example. Indeed, for an "and"-operation, we have $f_\&(1, a) = a$ and $f_\vee(a, 1) = 1$ for all a. In particular, for $b = c = 1$, we have $f_\vee(b, c) = 1$, $f_\&(a, b) = a$, and $f_\&(a, c) = a$. Thus, the left-hand side of the distributivity equality is equal to $f_\&(a, f_\vee(b, c)) = f_\&(a, 1) = a$, while the right-hand side is equal to $f_\vee(f_\&(a, b), f_\&(a, c)) = f_\vee(a, a)$. Thus, for $b = c = 1$, distributivity implies that $f_\vee(a, a) = a$ for every a.

It is easy to show that that the only "or"-operations satisfying this condition is $f_\vee(a, b) = \max(a, b)$. Indeed, if $b \le a$, then from the known property $f_\vee(a, 0) = a$, new property $f_\vee(a, a) = a$, and monotonicity $a = f_\vee(a, 0) \le f_\vee(a, b) \le f_\vee(a, a) = a$, we conclude that $f_\vee(a, b) = a$. Similarly, for $b \ge a$, we get $f_\vee(a, b) = b$. In both cases, we get $f_\vee(a, b) = \max(a, b)$.

So, if we want to require distributivity and still allow "or"-operations which are different from maximum, we should limit distributivity – at least to cases when $f_\vee(b, c) < 1$. In other words, we require that if $f_\vee(b, c) < 1$, then $f_\&(a, \max(b, c)) = \max(f_\&(a, b), f_\&(a, c))$. In the following text, this is how we will understand distributivity of "and"- and "or"-operations.

Comment. When $f_\vee(b, c) < 1$, then both sides of the distributivity equality are smaller than 1.

Indeed, due to monotonicity, we have $f_\&(a, f_\vee(b, c)) \le f_\&(1, f_\vee(b, c)) = f_\vee(b, c) < 1$, so the left-hand side is indeed smaller than 1.

Due to monotonicity, $f_\&(a, b) \le f_\&(1, b) = b$ and $f_\&(a, c) \le f_\&(1, c) = c$. Thus, due to monotonicity, we have $f_\vee(f_\&(a, b), f_\&(a, c)) \le f_\vee(b, c) < 1$. So, the right-hand side is also smaller than 1.

A second example of distributive operations. It is easy to come up with an example of "and"- and "or"-operations which are distributive in this sense. The notion of distributivity started with arithmetic, where multiplication is distributive with respect to addition: $a \cdot (b + c) = a \cdot b + a \cdot c$. It is therefore reasonable to consider an example, in which the "and"-operation is multiplication and the "or"-operation is addition. Multiplication $f_\&(a, b) = a \cdot b$ ("algebraic product") is indeed one of the most frequently used "and"-operations. In contrast, pure addition $a + b$ cannot be an "or"-operation, since:

- an "or"-operation, given two values $a, b \in [0, 1]$, should always return a value $f_\vee(a, b) \in [0, 1]$,
- while for numbers $a, b \le 1$, the sum $a+b$ can be larger than 1: e.g., when $a = b = 1$, we have $a+b = 2 > 1$.

Once we restrict the sum to 1 from above, i.e., consider the operation $f_\vee(a, b) = \min(a + b, 1)$, then we already get one of the most frequently used "or"-operations. If this case, if we limit ourselves to situations when the "or"-operation coincides with addition, i.e., when $b + c < 1$, then $f_\vee(b, c) = b + c$, so the left-hand side of the desired equality takes the form

$$f_\&(a, f_\vee(b, c)) = a \cdot f_\vee(b, c) = a \cdot (b + c).$$

For the right-hand side, we get $f_\&(a, b) = a \cdot b$ and $f_\&(a, c) = a \cdot c$. From $b + c \le 1$ and $a \le 1$, it follows that $a \cdot (b + c) = a \cdot b + a \cdot c \le 1$. Thus, we have

$$f_\vee(f_\&(a, b), f_\&(a, c)) = f_\vee(a \cdot b, a \cdot c) = a \cdot b + a \cdot c.$$

The equality between the expressions for the left-hand side and the right-hand sides now follows from the well-known fact that multiplication is distributive with respect to addition.

III. How to Approximate a Fuzzy Relation by Fuzzy Rules: Case of Distributive "and"- and "or"-Operations

Now that we explained why it is reasonable to require that the "and"- and "or"-operations are distributive, let us analyze how, under this requirement, we can approximate a given fuzzy relation by fuzzy rules. To explain this, let us first analyze this problem.

Different types of "or"-operations: reminder. Some of the "or"-operations are *Archimedean*, in the sense that for every two values $a \in (0, 1)$ and $b \in (0, 1)$ for which $f_\vee(a, a, \ldots, a)$ (n times) $> b$. A typical example of an Archimedean "or"-operation is the "algebraic sum" $f_\vee(a, b) = a + b - a \cdot b$.

For such operations, if $b < 1$ and $c < 1$, then we have $f_\vee(b, c) < 1$. In this case, we would have distributivity for all $b < 1$ and $c < 1$ and thus, by continuity, for all b and c, and we already know that this is only possible for $f_\vee(a, b) = \max(a, b)$ – which is not an Archimedean "or"-operation. Thus, if we want to require distributivity, we need to consider non-Archimedean "or"-operations.

A standard example of such an operation is an operation which is isomorphic to $f_\vee(a, b) = \min(a+b, 1)$, i.e., an operation of the type $f_\vee(a, b) = f^{-1}(\min(f(a)+f(b), 1))$, for some strictly increasing function $f(a)$. It is known (see, e.g., [3], [4]) that every "or"-operation is isomorphic to a lexicographic combination of such operations, Archimedean operations, and $\max$. It is also known that every "or"-operation, for any $\varepsilon > 0$, can be approximates, with accuracy ε, by an operation isomorphic to $f_\vee(a, b) = \min(a + b, 1)$.

It is reasonable to assume that the actual "or"-operation is isomorphic to $f_\vee(a, b) = \min(a+b, 1)$. As we have mentioned earlier, the main purpose of "or"-operation $f_\vee(a, b)$ is to estimate the expert's degree of belief $d(A \vee B)$ in a composite statement $A \vee B$ as $f_\vee(d(A), d(B))$. It is therefore reasonable to select an "or"-operation for which, for all the pairs of statements (A_k, B_k) for which we know both the degrees $d(A_k)$ and $d(B_k)$ *and* the actual expert's degree of belief $d(A_k \vee B_k)$, we should have $d(A_k \vee B_k) \approx f_\vee(d(A_k), d(B_k))$.

Because of the approximate character of an "or"-operation, we can always replace it with a very close one without changing the practical accuracy of the approximation. For example, if an estimate $f_\vee(d(A_k), d(B_k))$ approximates the actual expert's degree $d(A_k \vee B_k)$ with an accuracy of 10%, then replace the corresponding "or"-operation with another operation $f'_\vee(a, b) \approx f_\vee(a, b)$ one which is 0.01-close (or even 0.001-close) to $f_\vee(a, b)$, we get, in effect, the exact same approximation accuracy.

Since every "or"-operation can be approximated by a one which is isomorphic to $f_\vee(a,b) = \min(a+b,1)$, it makes sense to use the approximating isomorphic operation instead of the original one. In other words, it makes sense to assume that the actual "or"-operation is isomorphic to

$$f_\vee(a,b) = \min(a+b,1).$$

This is what we will assume in the following text.

How to describe distributive pairs of "and"- and "or"-operations. Under the above assumption, let us now describe all distributive pairs of fuzzy logic operations.

The fact that the "or"-operation is isomorphic to $f_\vee(a,b) = \min(a+b,1)$ means that if we "re-scale" all the original degrees of belief $a,b,c \in [0,1]$ into values $a' = f(a)$, $b' = f(b)$, and $c' = f(c)$, then the original relation $c = f_\vee(a,b)$ takes a simplified form $c' = \min(a'+b',1)$.

We can apply the same re-scaling to the "and"-operation $f_\&(a,b)$, resulting in a new "and"-operation $g(a',b') \stackrel{\rm def}{=} f(f_\&(f^{-1}(a'), f^{-1}(b')))$. One can easily check that this is indeed an "and"-operation (i.e., a t-norm). In the new scale, the distributivity condition takes the following form: if $b'+c' < 1$, then $g(a',b'+c') = g(a',b') + g(a',c')$. In other words, for each a', the function $b' \to g(a',b')$ is a monotonic additive function of b'.

It is known [1] that all monotonic additive functions have the form $f(x) = k \cdot x$. Thus, we have $g(a',b') = k(a') \cdot b'$ for some $k(a')$. Since every "and"-operation is commutative $g(a',b') = g(b',a')$, we get $k(a') \cdot b' = k(b') \cdot a'$. Dividing both sides of this equality by $a' \cdot b'$, we conclude that

$$\frac{k(a')}{a'} = \frac{k(b')}{b'}.$$

In other words, we conclude that the ratio

$$\frac{k(a')}{a'}$$

has the same value for all possible values $a' \in [0,1]$ – in other words, we conclude that this ratio is a constant. Let us denote this constant by r. Then, from

$$\frac{k(a')}{a'} = r,$$

we conclude that $k(a') = r \cdot a'$. Therefore, $g(a',b') = k(a') \cdot b' = r \cdot a' \cdot b'$. From the requirement that $g(1,1) = 1$, we conclude that $r = 1$ and thus, $g(a',b') = a' \cdot b'$.

Thus, each distributive pair is isomorphic to the above example of the "or"-operation $f_\vee(a,b) = \min(a+b,1)$ and the algebraic product $f_\&(a,b) = a \cdot b$.

Approximating a fuzzy relation by fuzzy rules: what we propose. Because of the above isomorphism, in the appropriate scale, the desired representation has the form

$$R(x'_1,\ldots,x'_n) = \sum_{r=1}^{n_r} d_r(x'_1,\ldots,x'_n),$$

where

$$d_r(x'_1,\ldots,x'_n) = A_{r1}(x'_1) \cdot \ldots \cdot A_{ri}(x'_i) \cdot \ldots A_{rn}(x'_n).$$

Thus, we get

$$R(x'_1,\ldots,x'_n) = \sum_{r=1}^{n_r} \prod_{i=1}^{n} A_{ri}(x'_i).$$

The problem of approximating a given function by expressions of this type is known as the problem of *tensor decomposition*. Many efficient algorithms have been developed for solving this problem; see, e.g., a recent survey [2] and references therein; many of these algorithms have been developed in the last few years.

We therefore propose to use these algorithms to approximate a given fuzzy relation by a sequence of fuzzy rules – and after using these algorithms, we should "re-scale" the resulting functions $A_{ri}(x'_i)$ back to the original scale, i.e., form functions $A'_{ri}(x_i) \stackrel{\rm def}{=} A_{ri}(f(x_i))$.

Comment. We are interested in representations with non-negative values $A_{ri}(x_i)$. Most tensor decomposition algorithms allow representations with functions of arbitrary sign, so we may end up with negative values of $A_{ri}(x_i)$.

This is OK if all we are interested in is approximation, but if we want an interpretable approximation, i.e., an approximation for which the values $A_{ri}(x_i)$ can be interpreted as membership functions, then we have to replace each negative value by the closest non-negative one, i.e., by 0.

It should be mentioned, however, that this replacement may somewhat decrease the approximation accuracy.

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation grants HRD-0734825 and HRD-1242122 (Cyber-ShARE Center of Excellence) and DUE-0926721.

REFERENCES

[1] J. Aczel, *Functional Equations and Their Applications*, Academic Press, New York, 1966.

[2] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank tensor approximation techniques", *GAMM-Mitteilungen*, 2013, Vol. 36, No. 1, pp. 53–78.

[3] G. Klir and B. Yuan, "Fuzzy Sets and Fuzzy Logic", Prentice Hall, Upper Saddle River, New Jersey, 1995.

[4] H. T. Nguyen and E. A. Walker, *A First Course in Fuzzy Logic*, Chapman and Hall/CRC, Boca Raton, Florida, 2006.

[5] C. Papadimitriou, *Computational Complexity*, Addison Welsey, Reading, Massachusetts, 1994.

[6] L. A. Zadeh, "Fuzzy sets", *Information and Control*, 1965, Vol. 8, pp. 338–353.