3. Fuzzy Constraint Satisfaction

3.1 Introduction

Traditional tools for formal modeling, reasoning, and computing are crisp, deterministic, and precise in character, but in many real world decision-making situations it is not possible to fulfill perfectly all conditions and objectives of a *constraint satisfaction problem* which models the environment that has to be optimized. Different reasons lead us to introduce a new concept for modeling constraint satisfaction problems in order to get a more suitable representation of real-world decision-making problems.

On the one hand, it is not always possible to formulate real-world conditions as numerical well defined, crisp constraints, because strict criteria for the validity of all system variables are not available. Most real-world application's knowledge is not hard, and the boundaries for satisfaction are of a softer kind. Considering production scheduling in industrial environments as a special CSP, we meet different types of *imprecision*, stemming from constraints that are blurred in definition and include *vagueness* and *uncertainty*. We can imagine that an operation on the schedule may start a „*little*" earlier and that „*small*" deviations of optimal values may be acceptable. In such a case classical CSP - methods would reject an efficient schedule even when the violations are insignificant with regard to the precision of the realistic limits of predictability. In this sense, it would be desirable to establish further functionality and flexibility in CSP-modeling and -solving techniques to handle *imprecision of data, vagueness of data, different importance of constraints, antagonistic criteria,* and the *relaxation of constraints* in order to solve an overconstrained CSP.

On the other hand, human knowledge and judgment gives values to imprecise variables as, e.g., for 'Speed': *very slow, slow, fast, very fast,* etc. As we see, it would be convenient to express such variables using *linguistic variables* [Zadeh75] definable operationally by mapping the preferred range of values of the parameters on to corresponding values of a membership function of the fuzzy set representing that variable. By a linguistic variable we mean a variable whose values are words or sentences in a natural or artificial language. For example, 'Speed' is a linguistic variable if its values are linguistic rather than numerical as we have seen above.

In the following sections we will introduce fuzzy set based constraints that help to model general multiple criteria optimization problems.

3.2 Fuzzy Set Theory

„As its name implies, the theory of fuzzy sets is, basically, a theory of graded concepts - a theory in which everything is a matter of degree or, to put it figuratively, everything has elasticity.“ [Zimmermann91].

In 1965 L. A. Zadeh presented the first publication on fuzzy sets, entitled „Fuzzy Sets“ [Zadeh65], in which he introduced concepts that

> „... provide a natural way of dealing with problems in which the source of imprecision is the absence of sharply defined criteria of class membership rather than presence of random variables.“ [Zadeh65]

3.2.1 An Introduction in Fuzzy Set Theory

Classical crisp sets have the property that an element only can either belong or not belong to a set. A fuzzy set is a class of objects with a continuum of grades of membership. Such a set is characterized by a membership (characteristic) function which assigns to each object a grade of membership. So the concept of fuzzy sets allows the elements of a given set to have various degrees of membership. Each element of the fuzzy set is mapped to a non negative real number $\in$ [0,1] by a membership function.

As we see, fuzzy set theory is a generalization of classical set theory, and so crisp sets can be seen as a special case of fuzzy sets, where the membership function provides only the two values 0 and 1 for membership and non membership. Formally, a fuzzy set A in X can be defined as a set of ordered pairs:

$$A = \{(x,\mu_A(x)) \mid x \in X\},$$

where X is a collection of objects denoted generically by x and $\mu_A(x)$ is the grade of *membership* of x in A.

EXAMPLE: The set A of „hot temperatures“:

$\mu_A(50° C)$	= 1	$\mu_A(25° C)$	= 0.5
$\mu_A(45° C)$	= 0.9	$\mu_A(10° C)$	= 0.1
$\mu_A(35° C)$	= 0.8	$\mu_A(5° C)$	= 0.01

3.2.2 Fuzzy Set Theoretic Operations

Operating on fuzzy variables makes it necessary to apply mathematically well-defined operations. At this time we should not forget that there is nothing fuzzy about fuzzy logic. In his article „Fuzzy Sets“ [Zadeh65] L. A. Zadeh presented for the first time the set theoretic operations on fuzzy sets, which basically are *equality, inclusion, complement, union, intersection, relation and convexity*, being extensions of classical set theory. As this operations are necessary for further computation, we will repeat the usual definitions in this section.

3.2.2.1 Equality

A fuzzy set A is equal to a fuzzy set B if

$$A = B \Leftrightarrow \mu_A(x) = \mu_B(x),\ \forall x$$

3.2.2.2 Inclusion

A fuzzy set A is included in a fuzzy set B if

$$A \subseteq B \Leftrightarrow \mu_A(x) \leq \mu_B(x),\ \forall x$$

3.2.2.3 Complement

A fuzzy set A’ is the complement of a fuzzy set A if

$$\mu_{A'}(x) = 1 - \mu_A(x),\ \forall x$$

3.2.2.4 Union

The union of two fuzzy sets A and B is the smallest fuzzy set C containing both A and B, whose membership function is defined by

$$\mu_C(x) = \max[\mu_A(x), \mu_B(x)]\ ,\ \forall x$$

3.2.2.5 Intersection

The intersection of two fuzzy sets A and B is the largest fuzzy set C which is contained in both A and B, whose membership function is defined by

$$\mu_C(x) = \min[\mu_A(x), \mu_B(x)] , \forall x$$

The min and max operators introduced by [Zadeh65] are not the only operators that can be used to model union and intersection, but the only ones that preserve such a structure for fuzzy sets.

3.2.2.6 Relation

An n-ary fuzzy set relation in X can be defined as a fuzzy set A in the product space $X_1 \times \ldots \times X_n$. The membership function for such a relation is of the form

$$\mu_A(x_{1,}\ldots,x_n.), \qquad x_i \in X_i, \qquad i = 1,\ldots,n$$

EXAMPLE: (taken from [Zadeh65]) The binary relation x >> y, x, y ∈ R (R denoting the real numbers), with the meaning „x is much greater than y“, which can be regarded as a fuzzy set in R × R. An appropriate membership function could have the following values

$\mu_A(10,5) = 0$
$\mu_A(100,10) = 0.7$
$\mu_A(100,1) = 0.9$
etc.

The composition of two fuzzy relations A and B, denoted by A ⊗ B, is defined as a fuzzy relation whose membership function is related to μ_A and μ_B by

$$\mu_{A \otimes B}(x,y) = \text{Sup Min } [\mu_A(x, z), \mu_A(z, y)],$$

where Sup Min denotes the greatest of all minima of $\mu_A(x, z)$ and $\mu_A(z, y)$.

3.2.2.7 Convexity

A fuzzy set A is convex (i.e. its membership function has only one peak) if and only if

$$\mu_A(\lambda x_1 + (1 - \lambda)\, x_2) \geq \min(\mu_A(x_1), \mu_A(x_2)), \qquad x_1, x_2 \in A, \lambda \in [0,1]$$

A basic property of convex fuzzy sets is, that the intersection of two convex sets is also convex.

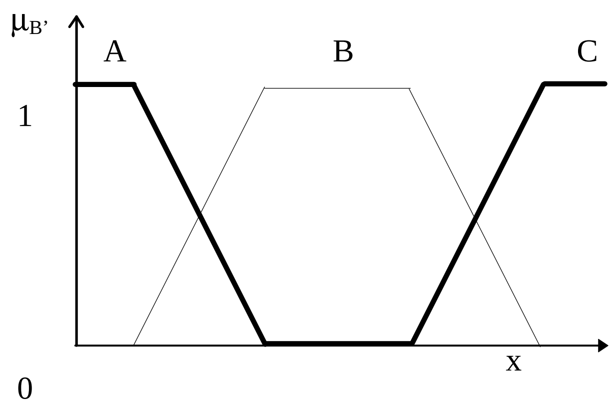

Figure 3.1: Complement $\mu_{\mathbf{B'}}$

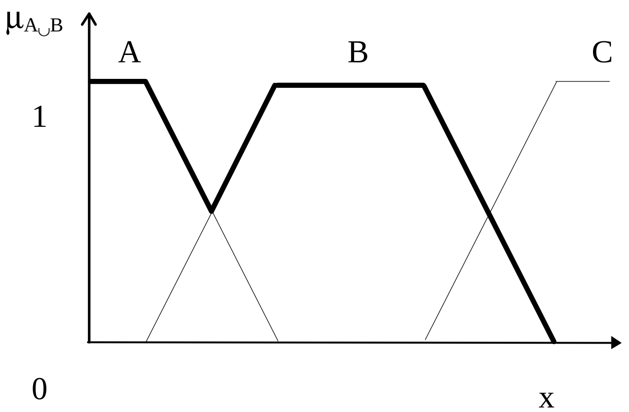

Figure 3.2: Union $\mu_{\mathbf{A\cup B}}$

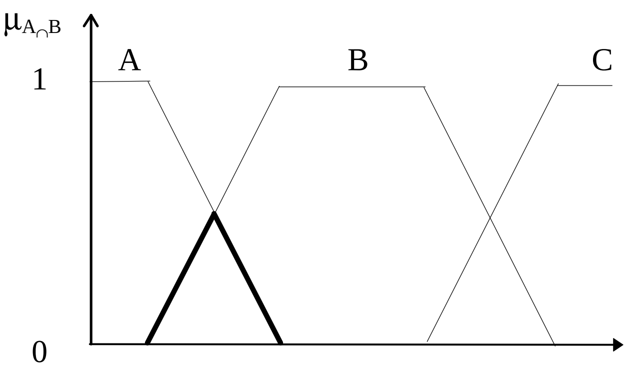

Figure 3.3: Intersection $\mu_{\mathbf{A\cap B}}$

3.3 Possibility Theory

The fact that in Fuzzy Constraint Satisfaction we have to handle variables whose actual values are uncertain led L. A. Zadeh to investigate on the relationships of fuzzy set theory to probability and possibility theory in his article „Fuzzy Sets as a Basis for a Theory of Possibility“:

> „Viewed in this perspective, a fuzzy restriction may be interpreted as a possibility distribution, with its membership function playing the role of a possibility distribution function, and a fuzzy variable is associated with a possibility distribution in much the same manner as a random variable is associated with a probability distribution.“ [Zadeh78]

We can imagine that in a job-shop scheduling problem some scheduling parameters like the duration of the tasks may be ill-known. Uncertainty stemming from the process can be represented by possibility theory. The specificity of a possibility distribution measures the degree to which the distribution allows one and only one element as its manifestation. As such it is a measure of amount of uncertainty or information. Detailed research on possibility theory has been done by L. A. Zadeh [Zadeh78], D. Dubois and H. Prade at Univeristé de Tolouse in „Possibility Theory“ [Dubois88] and R. Yager [Yager91].

[Yager91] defines possibility and specificity as follows:

Assume V is a variable taking its value in the set X. A proposition in the theory of approximate reasoning is a statement of the form

V is A

where A is a fuzzy subset of X. As suggested by [Zadeh78] such a proposition induces a possibility distribution Π defined on the space X. In particular for each $x \in X$, $\Pi(x) = \mu(x)$, where $\mu(x)$ is the membership grade of x in A. In this environment $\Pi(x)$ indicates the degree to which x is a possible value of V. The concept of a possibility distribution is in the spirit of a probability distribution. These two kinds of distributions both provide information about a variable whose actual value is uncertain. Yager introduced the *specificity of* Π to measure the amount of information contained in a possibility distribution, analogous to the Shannon entropy in probability distribution. The specificity of a possibility distribution measures the degree to which the distribution allows one and only one element as its manifestation.

To illustrate the relation between the concept of a fuzzy restriction and that of a possibility distribution we will give a brief example taken from [Zadeh78]:

EXAMPLE: Consider a numerical age, say u = 28, whose grade of membership in the fuzzy set *young* is approximately 0.7. First, we interpret 0.7 as the degree of compatibility of 28 with the concept labeled young. Then, we postulate that the proposition „John is young“ converts the meaning of 0.7 from the degree of compatibility of 28 with young to the degree of possibility, that John is 28 given the proposition „John is young“. In short, the compatibility of u with young comes converted into the possibility of that value of u given „John is young“.

In this thesis we will only discuss the motivation for the introduction of probability and possibility theory in fuzzy set theory. For detailed description please refer to [Zadeh78], [Dubois88], and [Yager91]. One important issue of these studies is the fact, that there are fundamental differences between the possibility and the probability of an event. Let $\pi_x(u)$ be the possibility distribution and $p_x(u)$ the probability distribution of the statement u, then we can interpret $\pi_x(u)$ as the ease with which the realization is u, while $p_x(u)$ is the probability determined by an observation of the investigated statement.

EXAMPLE: We consider the statement „John watched x hours TV“, x = {1,2,...,6} and the associated possibility and probability distributions.

x	**1**	**2**	**3**	**4**	**5**	**6**
$\boldsymbol{\pi_x(u)}$	1.0	1.0	1.0	0.8	0.6	0.4
$\mathbf{p_x(u)}$	0.2	0.7	0.1	0.0	0.0	0.0

Table 3.1: possibility and probability distributions

The sum of the probabilities of the single events is always 1, the sum of the possibilities is not determined:

$$P(X) = \sum p(x) = 1$$
$$\Pi(X) = \sum \pi(x) = \text{not determined}$$

3.4 Linguistic Variables

As mentioned in 3.1, in practical applications usually arises the necessity to handle imprecise variables that are to be represented as close to the nature as possible, because numerical information may not be available at all. In [Zadeh75] the author defines *humanistic systems* to be :

> „...system(s) whose behaviour is strongly influenced by human judgement, perceptions or emotions. Examples of humanistic systems are: economic systems, political systems, ...“

To be able to handle variables as, e.g., *low, high, very high*, etc., which are rather words than numerical data, L.A. Zadeh introduces *linguistic variables* which he defines as follows:

> „a linguistic variable is characterized by a quintuple *(X, T(X),U,G,M)* in which X is the *name of the variable*, T(X) is a *term set* of X, that is, the collection of its linguistic values; U is a *universe of discourse*; G is a *syntactic rule* which generates the terms in T(X); M is a *semantic rule* which associates with each linguistic value X its *meaning*, M(X), where M(X) denotes a fuzzy subset of U.“

The motivation for the use of words or sentences rather than numbers is that linguistic characterizations are, in general, less specific than numerical ones, and thus more suitable to describe human reasoning.

The following example should illustrate the concept of a linguistic variable. A value such as *young* can be seen as a fuzzy restriction upon a base variable. It is characterized by a *compatibility function*, which is much the same as a *membership function*.

EXAMPLE: Let *Age* be a *linguistic Variable* with its elements being the *linguistic terms {very young, young, old, very old}*. Due to human the life-expectancy the universe of discourse is U = [0,100]. The base variable u is the age in years of life. Figure 3.4 shows the membership functions of the linguistic terms.

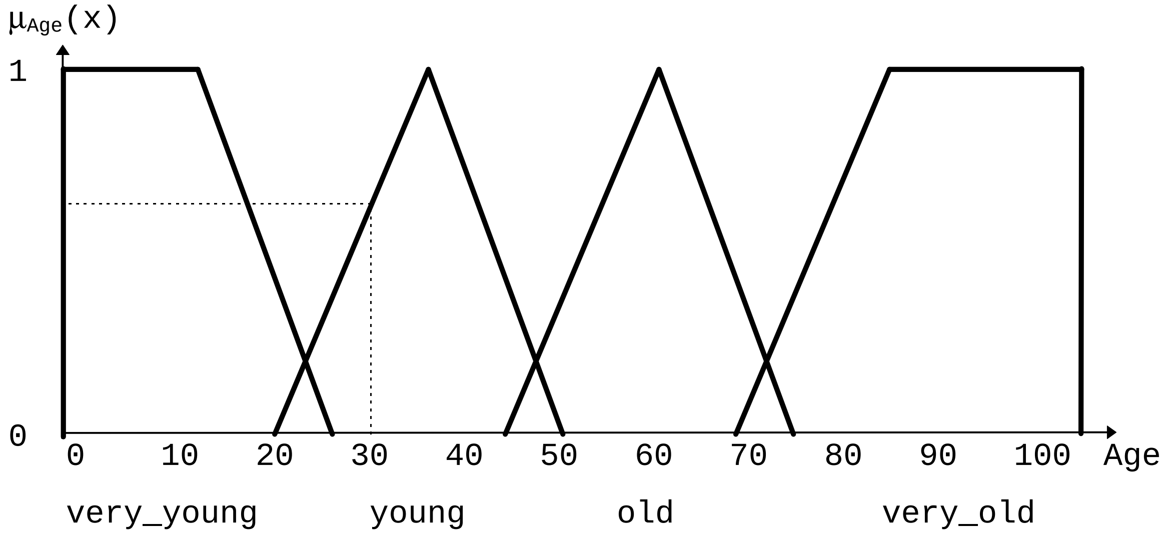

Figure 3.4: membership functions of the terms of the linguistic variable Age

For instance, the compatibility of Age = 30 with the linguistic term young is 0.7

3.4.1 Determining membership functions

A linguistic variable is usually intended to model human cognitive states. The determination of the appropriate modeling method for the membership functions for the linguistic terms of variable is a central point for a successful representation of human knowledge. We will take a short look at the most frequently types of membership functions.

- *bent*: the shape of such a function may be sigmoid, exponential, logarithmic or similar.

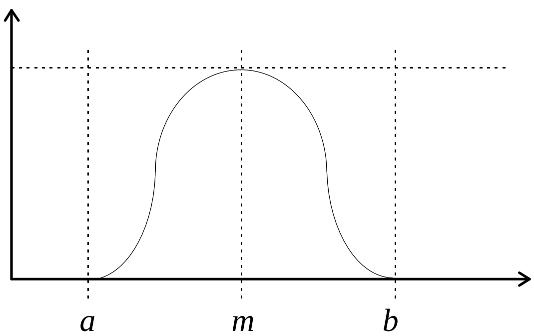

Figure 3.5: sigmoid membership function

❑ *triangular*: the triangle may be equilateral or not

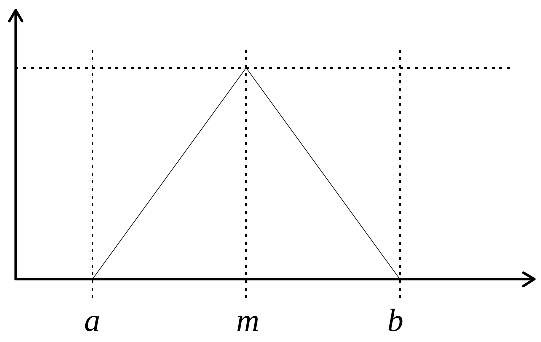

Figure 3.6: triangular membership function

❑ *trapezoid*: this form of membership appears in many applications. *start* is the point beyond membership is said to begin. In the core [c_{begin},c_{end}] the membership is definite. Then it declines until it reaches zero at*end*.

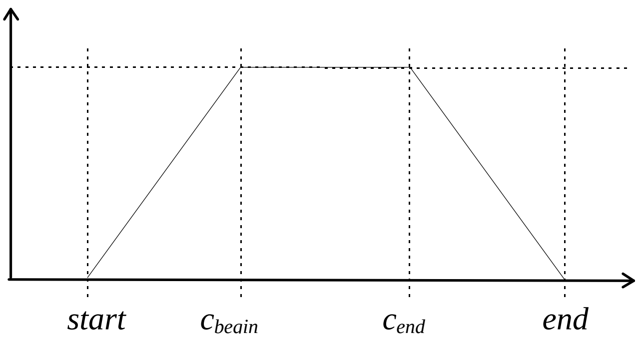

Figure 3.6: trapezoid membership function

❑ *ad hoc*: the shape may be of any representable form. Due to a special behavior of a system variable it is commonly drawn by experts.

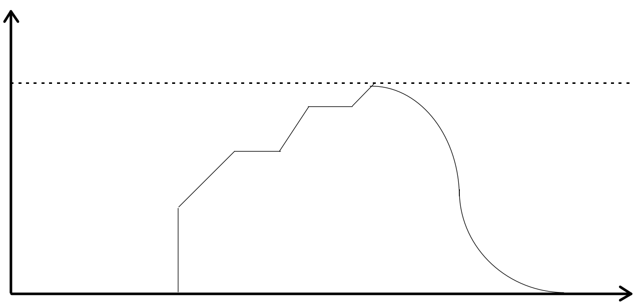

Figure 3.7: expert-drawn membership function

3.5 Fuzzy Constraints

Some constraints from certain application domains are often vaguely specified and are candidates for being reformulated as *fuzzy constraints*. Additionally, fuzzy constraints are well equipped for their use in repair based constraint satisfaction algorithms. In combination with heuristics it is possible to find suboptimal but good solutions which often are acceptable, while it is usually impossible to find an overall best solution for a given problem.

Like a cost-function with side-conditions in optimizing methods like the simplex algorithm in operations research, constraints are mathematical objects used to make explicit the logic behind a problem. They are used in decision making problems, e.g., design, configuration, planning, or scheduling. The following modeling and representation of fuzzy constraints are described in detail in [Dubois88a], [Dubois93], [Fargier93], and [Slany94].

[Slany94] introduces the following definitions:

A crisp k-ary constraint C_{crisp} of variables $x_1,...,x_k \in D_1\times...\times D_k$ can be formalized as a relation R_{crisp} with its characteristic function $\mathbf{1}_{crisp}$.

$$\mathbf{1}_{crisp}: \quad D_1\times...\times D_k \to \{0,1\}$$
$$(x_1,...,x_k) \to \mathbf{1}_{crisp}(x_1,...,x_k)$$

Obviously, only those k-tuples being assigned 1 from R_{crisp} are constraint-satisfying.

A soft k-ary constraint C_{soft} of variables $x_1,...,x_k \in D_1\times...\times D_k$ can be formalized as a relation R_{soft} with its characteristic function μ_{soft}.

$$\mu_{soft}: \quad D_1\times...\times D_k \to [0,1]$$
$$(x_1,...,x_k) \to \mu_{soft}(x_1,...,x_k)$$

R_{soft} assigns a fuzzy membership value $\mu_{soft}(x_1,...,x_k)$ from [0,1] to each k-tuple $(x_1,...,x_k)$, where μ_{soft} represents the level of preference between different tuples, a value of 1 means that all constraints are perfectly satisfied, and a value of 0 indicates that the instantiation of $(x_1,...,x_k)$ doesn't match at all.

Fuzzy modeled soft constraints provide a variety of features, as e.g.:

- defining soft constraints gives us the possibility to measure the satisfaction of constraints.

- in case of *underconstrained problems* we can specify preferences in the valid domain of the variables in order to grade solutions that otherwise would be all equal.

- in case of *overconstrained problems* we can specify margins for constraints where they can be relaxed while not leading to unacceptable results.

- fuzzy constraints of *different importance* can be modeled by specifying priorities among constraints.

- in combination with heuristics it is possible to find suboptimal but good solutions, which often are acceptable while it is usually impossible to find an overall best solution for a given problem.

- the propagation of uncertain values as possibility distributions.

EXAMPLE: A candidate for a quiz - show should meet the requirements to be young, intelligent and to watch television frequently. A rule-based fuzzy expert system should select one candidate from a set of competitors. A typical, slightly simplified rule looks as follows:

if	*candidate.IQ*	**is**	HIGH	**and**
	candidate.AGE	**is**	YOUNG	**and**
	candidate.TV	**is**	FREQUENTLY	
then	*candidate*	**is**	*selected*	

Here we can see again the use of the linguistic terms such as HIGH, YOUNG, and FREQUENTLY. What this terms correspond regarding the real age, IQ and what is meant by watching frequently TV, is described by appropriate membership-functions as we have seen in 3.4. As one can see, the restrictions for being a candidate are well equipped for being formulated as fuzzy constraints because the terms young, etc. are blurred in definition. Lets assume, that the following two candidates were selected by the system:

Candidate_1:	AGE	=	21 years
	IQ	=	112
	TV	=	6 hours

Candidate_2:	AGE	=	29 years
	IQ	=	128
	TV	=	2 hours

Both, Candidate_1 and Candidate_2 seem to meet the requirements. But the fact that only one candidate should be selected, leads us to question, which of them is the better one? To measure the „overall"-satisfaction of the three constraints we meet the necessity to

- *aggregate* constraints adequately
- introduce *priorities* among constraints

3.5.1 Aggregating Fuzzy Constraints

To solve a general constraint satisfaction problem we have to satisfy several constraints with one instantiation of all variables. In classical constraint satisfaction problems the solution of a multiple criteria optimization problem is obtained by taking the intersection of all relations, i.e. calculating the conjunction of all constraints. Any constraint violation results in a total rejection of the instantiation. However, in real-world applications, constraints may not have the same significance. For soft constraints the same concept as in classical CSPs is adapted by choosing a suitable *fuzzy conjunction operator*. The classical method for obtaining an order value for instantiations of a fuzzy constraint satisfaction problem is the *conjunctive combination* of all involved constraints. [Zadeh65] proposed the use of the *minimum* operator, but since then a variety of other operators has been investigated. In particular, the use of *t-norms* (triangular norms) as conjunctions is of interest because of their pleasing algebraic structure. On the other hand, human aggregation procedures in decision making environments have been analyzed by [Zimmermann91], and it has been shown that humans are able to perform trade-offs between conflicting goals when compensation is permitted. In multiple criteria optimization, this is often a requested feature, and various solutions have been developed. The following notations are taken from [Slany94]:

A t-norm is a binary operator $T:[0,1]^2 \to [0,1]$ such that for all $a,b,c,d \in [0,1]$:

$T(a,1) = a$	*(neutral element)*
$a \le b$ and $c \le d \Rightarrow T(a,c) \le T(b,d)$	*(monotonicity)*
$T(a,b)=T(b,a)$	*(commutativity)*
$T(Ta,b),c)=T(a,T(b,c))$	*(associativity)*

Because of the associativity property t-norms can be extended to a n-tuple by recursively applying the t-norm operator, what is necessary for the fuzzy conjunction of constraints.

Some basic t-norms with $a, b \in [0,1]$ are by definition

minimum:	$T_M(a,b) = \min(a,b)$
algebraic product:	$T_P(a,b) = a \cdot b$
Lukasiewicz:	$T_L(a,b) = \max(0,a+b-1)$
drastic product:	$T_W(a,b) = \min(a,b)$ if $\max(a,b) = 1$ and 0 otherwise

Other operator families can be found in [Zimmermann91] and [Yager88].

What t-norm operators T do for the intersection is done by the t-conorm operator S for the union, corresponding to the disjunction in classical logic. Any t-conorm can be derived from its associated t-norm as follows:

$$S(a,b) = 1 - T(1 - a, 1 - b)$$

Although t-norms and t-conorm are defined as binary operators they can be extended, via their associative property, for multiple operands, i.e. for an ‘and’-type aggregation operator for multiple constraints. The evaluation function E for an instantiation $d = (d_1,...,d_m)$ and constraints $(C_1,...,C_n)$ can be written as

$$E(d) = T(\mu_{C1}(d),...,\mu_{Cn}(d)),$$

$\mu_{Ci}(d)$ being the membership function of the i-th constraint. In [Yager88] the author shows that

$$T(a,b) \leq \min(a,b)$$

holds for any t-norm T and a,b ∈ [0,1]. From this and the associativity law for t-norms follows that in multiple criteria decision making the use of t-norms for 'anding' allows no compensation for one weak satisfaction:

$$T(a_1,...,a_n) \leq \min(a_1,...,a_n)$$

For t-conorms the following 'or'-style aggregation function E'(d) is the analogon of t-norms *and* functions.

$$E'(d) = T(\mu_{C1}(d),..., \mu_{Cn}(d))$$

Similarly

$$S(a,b) \geq \max(a,b)$$
$$S(a_1,...,a_n) \geq \max(a_1,...,a_n)$$

holds for any t-conorm S and a, b ∈ [0,1], which points out the characteristic of the *or*-type aggregation of t-conorms: In contrary to E(d), which does not allow the distraction from one individual bad satisfaction, E'(d) does not allow the distraction from one good satisfaction.

In [Yager88] Yager proposes another type of operator called an <u>o</u>rdered <u>w</u>eight <u>a</u>veraging (OWA) operator. In practical applications one may desire to have an operator, which neither leads an instantiation being dragged down by one single weak satisfaction, nor an instantitation being pulled up by one high satisfaction degree. The general definition given for this n-ary operator, which lies somewhere between *and* and *or*, is introduced in [Yager88] as follows:

Assume $a_1,...,a_n, W_1,...,W_n \in [0,1]$,
b_j the jth largest element in the collection $a_1,...,a_n$, and

$$\sum_{i=1}^{n} W_i = 1$$

then the OWA operator is defined as follows:

$$OWA_W(a_1, \ldots, a_n) = \sum_{i=1}^{n} W_i \cdot b_i$$

For instance, by specifying the weight vector

$$\vec{W}_{avg} = \begin{pmatrix} 1/n \\ \vdots \\ 1/n \end{pmatrix}$$

the normal averaging operator is obtained. Similarly, the maximum and minimum operators can be defined by the appropriate weighting vectors. For any t-norm T and t-conorm S Yager observed the following relations:

$$\underbrace{T_W \leq T \leq T_M}_{and-type} \leq \underbrace{OWA}_{and/or-type} \leq \underbrace{S_M \leq S \leq S_W}_{or-type}$$

where S_W denotes the t-conorm associated to T_W.

The weights are associated with a particular ordered position b_j rather than a particular d_j. The OWA operator therefore behaves invariant with respect to permutations of the input parameters. Furthermore, all OWA operator fulfill the idempotency property, which, in the case of idempotent satisfaction degrees d_j results in an overall satisfaction degree of the same extent. This means that all constraints are equally important. From the families of t-norms and t-conorms only T_M and S_M (being minimum and maximum respectively) accomplish the idempotency property.

OWA operators increase when aggregating additional better satisfaction degrees, decrease when aggregating additional worse satisfaction degrees, and remain equal when aggregating additional equal satisfaction degrees. Thus the behavior of OWA operators is often better suited for multiple criteria decision making in real-world situations where multistage inference steps are necessary, in order not to dilute knowledge excessively.

3.5.2 Fuzzy Constraints of different importance

The example in 3.5 has shown that in real-world applications often criteria do not all have the same importance. Thus it is reasonable to introduce relative priorities of constraints when instantiations are evaluated and compared. Additional information about preferences between constraints can be achieved by the consideration of weighting vectors that describe the relative importance of a constraint. Note that this weighting vector is not to be confused with the OWA weighting vector we have seen in the previous section.

[Yager81] argues that ranking or weighting of objectives can be achieved by linear orderings, intervals, relative ratio, or absolute ratings. Fargier et. al. [Fargier93] propose to order constraints with respect to each other by giving them a priority degree. A coefficient $w \in [0,1]$ is attached to each constraint, with a higher w indicating a comparatively higher importance. These priorities are transformed into constraint satisfaction degrees

$$\mu_{soft,w}(x_1,...,x_k) = \max(1 - w, \mu_{soft}(x_1,...,x_k))$$

This ensures that w represents the degree to which a constraint *must* be satisfied. The number (1 - w) measures to what extent it is possible to violate a constraint.

Yagers approach is slightly different. He does not use numbers but a finite set of ordered symbols to represent the weights of the criteria and defining the semantic of the '$\neg$' operator in an obvious way (e.g. $\neg$high = low, ...). Such an ordering could for instance be

$$\text{perfect} \geq \text{very high} \geq \text{high} \geq \text{medium} \geq \text{low} \geq \text{very low} \geq \text{lowest}$$

The difference between Yager's method and the method specified by Fargier et al. is more of terminological than semantic nature, as we can see:

$$\vec{W}_{FARGIER} = \begin{pmatrix} 0.3 \\ 0.2 \\ 1 \\ 0.8 \\ 0.1 \end{pmatrix} \qquad \vec{W}_{YAGER} = \begin{pmatrix} low \\ very\_low \\ perfect \\ high \\ lowest \end{pmatrix}$$

A detailed description can be found in [Slany94], [Yager81] and [Fargier93].

3.6 Conclusion

In this section we gave a short introduction in fuzzy set theory and possibilistic theory. We developed fuzzy constraints used in partial constraint satisfaction for real-world multiple criteria decision making. We introduced fuzzy constraints of different importance that allow compromising between antagonistic criteria, as well as some basic concepts for aggregating several constraints. The most commonly used operators, namely t-norms and t-conorms, the OWA operator and a weighting scheme have been described without getting lost in details in order to get the reader prepared for the following sections. A detailed description of these concepts can be found in [Dubois88a], [Dubois93], [Fargier93], [Yager81] [Zadeh65], and [Zimmermann91]. In the next section we will briefly restate the most common and most important definitions of what scheduling is and where it can be used for.