New Developments in Fuzzy Sets, Intuitionistic Fuzzy Sets, Generalized Nets and Related Topics. Volume II: Applications

New Developments in Fuzzy Sets, Intuitionistic Fuzzy Sets, Generalized Nets and Related Topics
Volume II: Applications

Editors

Krassimir T. Atanassov
Władysław Homenda
Olgierd Hryniewicz
Janusz Kacprzyk
Maciej Krawczak
Zbigniew Nahorski
Eulalia Szmidt
Sławomir Zadrożny

SRI PAS IBS PAN

New Developments in Fuzzy Sets, Intuitionistic Fuzzy Sets, Generalized Nets and Related Topics Volume II: Applications

Systems Research Institute
Polish Academy of Sciences

New Developments in Fuzzy Sets, Intuitionistic Fuzzy Sets, Generalized Nets and Related Topics
Volume II: Applications

Editors

Krassimir T. Atanassov
Władysław Homenda
Olgierd Hryniewicz
Janusz Kacprzyk
Maciej Krawczak
Zbigniew Nahorski
Eulalia Szmidt
Sławomir Zadrożny

IBS PAN **SRI PAS**

Systems Research Institute
Polish Academy of Sciences
Newelska 6, 01-447 Warsaw, Poland
www.ibspan.waw.pl

ISBN 83-894-7541-3

Dedicated to Professor Beloslav Riečan on his 75th anniversary

Aggregation of bipolar satisfaction degrees

Tom Matthé and Guy De Tré
Department of Telecommunications and Information Processing
Ghent University
Sint-Pietersnieuwstraat, B-9000 Ghent, Belgium
{Tom.Matthe;Guy.DeTre}@UGent.be

Abstract

In recent years, there has been a growing interest in dealing with user preferences expressing both positive and negative information in flexible querying of databases. This is what is usually referred to as bipolar querying of databases. Moreover, the positive and negative preferences do not necessarily have to be each other's complement, what is referred to as heterogeneous bipolarity. Bipolar Satisfaction Degrees have been introduced as a framework, using an independent satisfaction and dissatisfaction degree, to deal with such heterogeneous bipolarity. In this paper, an overview of different ways to aggregate these Bipolar Satisfaction Degrees is given, possibly taking into account weights allowing to make distinctions in relative importance between the different preferences.

Keywords: flexible querying, bipolar querying, bipolar satisfaction degrees, aggregation.

1 Introduction

It can be observed that in daily communication, people tend to use both positive and negative statements when expressing their preferences. Positive statements are used to express what is possible, satisfactory, permitted, desired or acceptable, whereas negative statements express what is impossible, unsatisfactory, not permitted, rejected, undesired or unacceptable. For example, consider the specification of user preferences in the context of car selection. Positive criteria are 'I prefer a white or blue car', 'I want a car with airbags', 'the car's average highway fuel consumption must be lower than 7 liters/100km' and 'I would like to buy a small car'. Examples of negative criteria are 'I don't want a black car', 'I will not buy a monospace car', 'a diesel engine is unacceptable', and 'I don't want a car that is older than 4 years'. In some situations, it is easier for a user to express negative statements while in other situations the use of positive statements is preferred by the user. Sometimes one can even have both a positive and a negative statement at the same time. This is especially the case when the user doesn't have complete control over the domain, or when the domain is too large to completely specify the satisfaction for every value in the domain, as is for example the case with available car colours. These observations illustrate the bipolar nature of natural language where bipolarity refers to the use of both positive and negative statements.

In some cases, positive and negative information have clear symmetric semantics and thus can be derived from each other. For example, if somebody wants to buy a car with airbags then it is usually clear that this person will not buy a car without airbags. However, there are also cases where it is not possible to make such strong conclusions. As an example, consider the situation where somebody does not want to buy a black car. This does not imply that this person will equally accept all car colours that are not black. Even combined with a preference for white or blue cars, there is still an underspecification of information which could be due to an indifference for some colours or simply to the inability to enumerate all car colours. This illustrates that bipolarity of information should in general be considered as having heterogeneous semantics. The naming 'heterogeneous bipolarity' is often used to denote such heterogeneous semantics [9]. In the presented heterogeneous bipolar approach, what is neither explicitly permitted, nor forbidden is considered to be unspecified, which could, among others, be due to indifference or hesitation of the user with respect to what is permitted or not, or due to the inability of the user to specify all (un)permitted values within the criteria. This consideration reflects the heterogeneous bipolar characteristic of the approach. Like in regular 'fuzzy' querying, elementary criteria can contain vague terms which are typical for natural language and are modeled by regular fuzzy sets [2, 3, 12, 16, 20], and which will result in a satisfaction degree in $[0, 1]$. But here, unlike in regular fuzzy querying, the elementary criteria will be used to express both what is desired, as well as what is undesired, independently of each other.

To deal with the heterogeneous bipolarity, a semantically richer query satisfaction modeling approach, which is more consistent with human reasoning, is needed, because regular 'fuzzy' database querying systems do not address this issue. Pioneering work in the area of heterogeneous bipolar database queries has been done in [13], which seems to be the first approach where a distinction was made between mandatory query conditions and desired query conditions which express just mere preferences. Desired and mandatory conditions can be viewed as specifying positive and negative information, respectively. Indeed, the inverse of a mandatory condition specifies what must be rejected and hence what is considered as being negative with respect to the query result. Whereas desired conditions specify what is considered as being positive. Later on, this idea has been incorporated in 'fuzzy' querying techniques. In the twofold fuzzy set based approach [8, 9], the user can specify in an elementary query condition which values are permitted with respect to the query result and which among these values are really desired. Alternatively, an approach where distinction has been made between required and preferred conditions and bipolar queries are represented by the fuzzy 'winnow' operator, has been presented in [19]. Bipolarity is then studied considering queries with preferences as in [13].

Other approaches, based on Atanassov intuitionistic fuzzy sets and departing from the specification of which values are desired and which values are undesired, are for example presented in [5, 6, 15]. In this paper, although other approaches are also possible, only this last approach is considered and a logical framework based on bipolar satisfaction degrees (BSDs) is used to handle query satisfaction. In this approach, the satisfaction degrees, as found in the regular 'fuzzy' querying, are extended with an extra, independent, dissatisfaction degree.

This paper will give an overview of different ways to aggregate BSDs. The remainder of the paper is structured as follows. In Section 2 some preliminaries on the proposed framework of bipolar satisfaction degrees are presented. The basic way to aggregate them, not taking into account any weights, are given in Section 3. Some statements, regardless of whether they are positive or negative, can be more important than others. Section 4 presents different ways to extend the basic aggregation techniques, supporting the use of weights to model the difference in importance. In more complex systems, like decision support systems, more advanced aggregation schema's can be necessary. Section 5 studies one way of how BSDs can be incorporated in such more advanced aggregation schema. Next, in Section 6 it is illustrated how a list of BSDs can be ranked, in order to find the best alternative for a query or decision. Finally, in the concluding Section 7, some general conclusions about the paper are presented.

2 Bipolar Satisfaction Degrees (BSDs)

Query satisfaction degrees $s \in [0, 1]$ in regular 'fuzzy' querying do not allow it to explicitly distinguish between heterogeneous satisfaction and dissatisfaction in query evaluation. Due to the fact that in standard approaches the negation operator is usually treated as an involutive operator (most often, $\neg x = 1 - x$), it is explicitly assumed that a record with satisfaction degree s does not satisfy the query to an extent $1 - s$. This assumption does not generally hold when dealing with heterogeneous bipolar query criteria specifications. If we consider negation as no longer being involutive, i.e., if we consider that the negation of a positive criterion specification is not generally implying the negative counterpart of that criterion, then we need a semantical richer framework that allows to explicitly and independently keep track of query dissatisfaction. Reconsidering the case where a user does not want to buy a black car, we can meaningfully assume that not all cars that are not black will be accepted by the user. There could be other car colours, although not explicitly specified, which could also be rejected by the user. Reversely, if the user is stating that he or she wants to buy a blue car, we can for similar reasons not exclude to the same extent all cars that are not blue. There could be other car colours, although not explicitly specified, which could also be acceptable for the user. To explicitly cope with such situations of heterogeneous bipolarity in query criteria evaluation and handling, a logical framework based on BSDs has been proposed [15].

Definition and Basic Characteristics

By definition, a bipolar satisfaction degree (BSD) is a pair $(s, d), s, d \in [0, 1]$ where s is called the satisfaction degree and d is called the dissatisfaction degree. Both s and d take their values in the unit interval [0,1] and are independent of each other: they independently denote to which extent the BSD respectively represents 'satisfied' and 'dissatisfied'. Extreme values for s and d are 0 ('not at all') and 1 ('fully'). As such and as special cases, the BSD (1,0) represents 'fully satisfied, not dissatisfied at all', whereas (0,1) represents 'not satisfied at all, fully dissatisfied'. The set of all possible BSDs will be denoted by $\tilde{\mathbb{B}}$, i.e., $\tilde{\mathbb{B}} = \{(s, d)|s, d \in [0, 1]\}$.

From a semantical point of view, BSDs are closely related to Atanassov intuitionistic fuzzy sets (AFS) [1], except that it is explicitly assumed that there is no consistency condition for BSDs, i.e., a condition like $0 \leq s + d \leq 1$ is missing. Indeed, because s and d are considered to be completely independent of each other, it is allowed that $s + d > 1$. The motivation for this is that BSDs try to reflect heterogeneous bipolarity in human reasoning, and that human reasoning can sometimes be inconsistent.

Two measures can be deduced from the bipolar pair (s, d):

1. $s + d$: this measure ($\in [0, 2]$) indicates how the BSD is specified. Three cases can be considered:

   - $s + d < 1$: the BSD is *underspecified*. In the context of the evaluation of a query criterion, this means that there is an amount, $1 - s - d$, of indifference (or hesitation, as it is called in the case of AFSs) about whether the criterion is satisfied or not.
   - $s + d = 1$: the BSD is *fully specified*. In fact, in this case it holds that $d = 1 - s$, so this is the case of regular, involutive query satisfaction modeling. One also speaks of symmetric bipolarity.
   - $s + d > 1$: the BSD is *overspecified* and denotes an amount, $s + d - 1$, of conflict.

2. $s - d$: this measure ($\in [-1, 1]$) can be used as a ranking value for BSDs. Three special cases can be distinguished:

   - $s - d = 1$: in this case it must be that $s = 1$ and $d = 0$, so this is the case of *full satisfaction* (without any hesitation or conflict).
   - $s - d = -1$: in this case it must be that $s = 0$ and $d = 1$, so this is the case of *full dissatisfaction* (without any hesitation or conflict).
   - $s - d = 0$: in this case the ranking is *neutral*. The criterion is as satisfied as it is dissatisfied.

   Remark that bipolar satisfaction degrees where $s = d$ have neutral ranking.

Remark that BSDs where $s = d$ have neutral ranking, with respect to the presented ranking value. Other ranking functions are also possible, e.g., assigning more importance to either the satisfaction degree or the dissatisfaction degree. Ranking functions will be described more elaborately in Section 6.
When using the ranking $s - d$, a consequence is that elementary query conditions cannot be seen as strict restrictions, but only as preferences or desires. This means that tuples of which the evaluation leads to a dissatisfaction degree $d = 1$, or dually a satisfaction degree of $s = 0$, should not a priori be excluded as being totally unsatisfactory. Indeed, e.g., the BSDs $(1, 1)$ and $(0, 0)$, although having $d = 1$ (respectively $s = 0$), both have neutral ranking ($s - d = 0$) and are hence situated in the middle of the ranking spectrum due to which they should not be catalogued as being totally unsatisfactory.

2.1 Bipolar Query Conditions

In what follows, the query conditions c and c_i are considered to be bipolar query conditions, i.e., they specify both positive and negative preferences at the same time, on the same attribute. In general, a bipolar query condition can specify both a satisfaction degree ($\in [0,1]$) and a dissatisfaction degree ($\in [0,1]$) for every value in the domain. Therefor, heterogeneous bipolar specifications of user preferences inside elementary query selection conditions can formally be modeled with a bipolar extension of fuzzy sets. *Atanassov (intuitionistic) fuzzy sets* (AFSs) [1] are an example of such an extension. An AFS F over a universe U is formally defined by

$$F = \{(x, \mu_F(x), \nu_F(x)) | (x \in U) \wedge (0 \leq \mu_F(x) + \nu_F(x) \leq 1, \forall x \in U)\} \quad (1)$$

where $\mu_F : U \to [0,1]$ and $\nu_F : U \to [0,1]$ are respectively called the membership and non-membership degree functions and $0 \leq \mu_F(x) + \nu_F(x) \leq 1$, $\forall x \in U$ reflects the consistency condition, which states that an AFS can not be overspecified.

Consider an elementary selection condition c_A on a (relational) database attribute A with domain dom_A which expresses the user's preferences related to the tuple values of A. Then, in its simplest general form c_A can be modeled by an AFS

$$c_A = \{(x, \mu_{c_A}(x), \nu_{c_A}(x)) | (x \in dom_A)\}. \quad (2)$$

The membership function μ_{c_A} defines the positive preferences of the user, i.e., the membership grade $\mu_{c_A}(x)$ associated with a domain value $x \in dom_A$ denotes to what extent x is considered to be satisfactory with respect to attribute A. The non-membership function ν_{c_A} defines the negative preferences of the user, i.e., the non-membership grade $\nu_{c_A}(x)$ associated with a domain value $x \in dom_A$ thus denotes to what extent x is considered to be unsatisfactory. To adequately reflect the real-world cases where user preferences can be overspecified the consistency condition for AFSs must even be relaxed, which implies that $\mu_{c_A}(x)+\nu_{c_A}(x) \leq 1$, $\forall x \in dom_A$ must not necessarily hold.

3 Basic Aggregation of BSDs

The evaluation of a bipolar query condition c_A results in a BSD $\tilde{b}^R_{c_A} = (s^R_{c_A}, d^R_{c_A})$ for every record R [15], where $s^R_{c_A}$ is equal to the membership grade $\mu_{c_A}(R_A)$ and $d^R_{c_A}$ is equal to the non-membership grade $\nu_{c_A}(R_A)$, with R_A the value of record R for attribute A. When evaluating entire queries, composed of n query conditions c_i, $i = 1, \ldots, n$, all individual BSDs must be aggregated to come up with a result for the entire bipolar query (also expressed by a BSD). The basic aggregation of BSDs is presented in the subsections below.

3.1 Conjunction

The result of the conjunction of two query conditions c_1 and c_2 is the intersection of the set of records satisfying c_1 with the set of records satisfying c_2. When the query conditions are expressed over the same attribute and hence the same domain, this can be translated to finding the set of records satisfying the intersection of the two query conditions. In case of bipolar query conditions, which are expressed by an AFS like concept (e.g., an Atanassov intuitionistic fuzzy set [1] without the consistency condition), the same technique as for the intersection of AFSs could be used. Suppose c_1 and c_2 are expressed by membership functions μ_{c_1} and μ_{c_2} and non-membership functions ν_{c_1} and ν_{c_2}, then the intersection of the two conditions, according to the AFS approach is

$$c_1 \wedge c_2 = \{(x, \min(\mu_{c_1}(x), \mu_{c_2}(x)), \max(\nu_{c_1}(x), \nu_{c_2}(x)))|x \in dom_A\} \quad (3)$$

with A being the attribute with domain dom_A, over which c_1 and c_2 are expressed.

This formula can directly be translated to the presented framework of BSDs: the BSD $(s^R_{c_1 \wedge c_2}, d^R_{c_1 \wedge c_2})$ of the conjunction of conditions c_1 and c_2,for the evaluation of a record R, can be calculated as follows:

$$(s^R_{c_1 \wedge c_2}, d^R_{c_1 \wedge c_2}) = (\min(s^R_{c_1}, s^R_{c_2}), \max(d^R_{c_1}, d^R_{c_2})) \quad (4)$$

Of course, this way of calculating the conjunction of two query conditions can only be done under the assumption that the two conditions are expressed over the same attribute and domain. If this assumption is not valid, the intersection of the fuzzy sets specifying the conditions cannot be performed. So, the argumentation for Equation (4) is only valid under the assumption that the two conditions are expressed over the same attribute and domain. However, Equation (4) still remains valid, even without this assumption. Intuitively, this can be seen as follows:

- For the conjunction to be satisfactory, both conditions must be satisfactory. Therefore the minimum of both individual satisfaction degrees is taken.
- For the conjunction to be unsatisfactory, one of both conditions must be unsatisfactory. Therefore the maximum of both individual dissatisfaction degrees is taken.

This approach is not exactly the same as in [8], where positive information is treated as mere desires and the fulfillment of one desire is enough for the whole to be desirable, which would mean taking also the maximum for the calculation of the satisfaction degree.

Besides the minimum and maximum, other aggregation operators based on triangular norms and co-norms, can also be used if a reinforcement effect is needed or desired.

3.2 Disjunction

The disjunction of two query conditions c_1 and c_2 can be treated dually to the conjunction. For a disjunction to be satisfactory, one of both conditions must be satisfactory. Therefore the maximum of both individual satisfaction degrees can be taken. On the other hand, for a disjunction to be unsatisfactory, both conditions must be unsatisfactory. Therefore the minimum of both individual satisfaction degrees can be taken. So, the BSD $(s^R_{c_1 \vee c_2}, d^R_{c_1 \wedge c_2})$ of the disjunction of conditions c_1 and c_2, for the evaluation of a record R, can be calculated as follows:

$$(s^R_{c_1 \vee c_2}, d^R_{c_1 \vee c_2}) = (\max(s^R_{c_1}, s^R_{c_2}), \min(d^R_{c_1}, d^R_{c_2})) \tag{5}$$

3.3 Negation

The BSD $(s^R_{\neg c}, d^R_{\neg c})$ of the negation of condition c, for the evaluation of a record R, can be obtained by switching the satisfaction degree and dissatisfaction degree of the BSD of the initial condition c:

$$(s^R_{\neg c}, d^R_{\neg c}) = (d^R_c, s^R_c) \tag{6}$$

In fact, the same effect of negation can also be achieved by just switching the membership and non-membership function expressing the query condition.

Remark that this is not the same as taking the inverse of the satisfaction degree and dissatisfaction degree of the BSD of the initial condition c. This would lead to $(s^R_{\neg c}, d^R_{\neg c}) = (1 - s^R_c, 1 - d^R_c))$, but gives the result of the case of total indifference (BSD (0,0)) and the case of total conflict (BSD (1,1)) being each others negation. This is incorrect since the negation of total indifference should still be total indifference (and the same for total conflict).

3.4 General Remarks

A few remarks can be made considering the aggregation technique presented above:

- The law of excluded middle does not hold, i.e., with x a BSD ($x \in \tilde{\mathbb{B}}$), $x \vee \neg x$ is not necessarily equal to the BSD $(1, 0)$. However, the law of excluded middle does not hold for the traditional framework of regular, involutive, query satisfaction degrees either.

- In case of symmetric bipolarity (i.e., $s + d = 1$), this "*symmetric*" property is kept under aggregation if all BSDs themselves are symmetric. E.g., for conjunction of conditions c_1 and c_2, with $s^R_{c_1} + d^R_{c_1} = 1$ and $s^R_{c_2} + d^R_{c_2} = 1$, it holds that

$$
\begin{aligned}
&(s^R_{c_1 \wedge c_2}, d^R_{c_1 \wedge c_2}) \\
&= (\min(s^R_{c_1}, s^R_{c_2}), \max(d^R_{c_1}, d^R_{c_2})) \\
&= \begin{cases} (s^R_{c_1}, d^R_{c_1}) & \text{if } s^R_{c_1} \leq s^R_{c_2} \\ & \text{because } d^R_{c_1} = 1 - s_{c_1} \geq d^R_{c_2} = 1 - s_{c_2} \\ (s^R_{c_2}, d^R_{c_2}) & \text{if } s^R_{c_2} < s^R_{c_1} \\ & \text{because } d^R_{c_2} = 1 - s_{c_2} > d^R_{c_1} = 1 - s_{c_1} \end{cases}
\end{aligned}
$$

  and as a result: $s^R_{c_1 \wedge c_2} + d^R_{c_1 \wedge c_2} = 1$

  In fact, in the symmetric case, the aggregation falls back to the aggregation of regular query satisfaction degrees. This makes it clear that the framework of BSDs is a true generalisation of the traditional approach using regular satisfaction degrees.

- One has to be careful when providing only positive or only negative information in the bipolar query condition, because in that case the non-membership function (respectively membership function) is assumed to be the inverse of the membership function (respectively non-membership function) specified. As a result, e.g. in case only positive information is given, the non-membership function will be 1, hence $d = 1$, for all values where the membership function is equal to 0 (either because it is explicitly specified or because the domain is not entirely known in advance). This makes that the conjunction with other conditions will always result is a BSD with $d = 1$ for these values, since in a conjunction the maximum of the dissatisfaction degrees is taken. So, in a conjunction, the impact of specifying only positive information is quite large for the values where the membership function is equal to 0. The final result will then always be a BSD with $d = 1$, making that it will always be in the bottom half of the ranking spectrum ($\in [-1, 0]$). On the other hand, when only negative information is given, the impact is not so big for the values with non-membership degree equal to 0 (again in case of conjunction). In that case, the satisfaction degree will be equal to 1 for these values, but since the minimum of the satisfaction degrees is taken for the conjunction, $s = 1$ is the neutral element of this operation. So, in case of conjunction, especially when the domain at hand is not completely known, it is safer to specify only negative information than to specify only positive information. Dually, in case of disjunction, the inverse holds and it is safer to specify only positive information than to specify only negative information, or one would end up with a final BSD with $s = 1$ for the values where the concerning non-membership function is equal to 0.

- Finally, one has to remark that there exist special cases which might seem strange. E.g. the conjunction of '*total indifference*' and '*total conflict*' results in '*total dissatisfaction*'. Indeed:

  $$(0, 0) \wedge (1, 1) = (0, 1)$$

  This might seem counterintuitive at first sight, but this is a result of the complete independence of the satisfaction degree and dissatisfaction degree respectively. A similar result is obtained for the disjunction: $(0, 0) \vee (1, 1) = (1, 0)$.

4 Weighted Aggregation

When expressing queries (bipolar or not), one way to model the difference in importance between different (bipolar) query conditions is by using weights. So, also in the presented framework of BSDs, it must be possible to deal with such weights. In the following subsections, different ways to deal with weights in aggregating BSDs are presented. The first approach uses the basic aggregation operators, presented in the previous section, as underlying aggregation operator, but performs a premodification step to take into account the impact of the weights. This approach is described in Subsection 4.1. In the other Subsection 4.2, the other approaches which are not based on the basic aggregation operators, but use averaging operators, are handled.

4.1 Premodification

In this approach, it is assumed that the importance of a criterion, with respect to the final result, depends only on the criterion itself, not on the degree to which the criterion is satisfied. So weights w_i can be attached to the individual conditions c_i, with $w_i \in [0, 1]$. The semantics of the weights are as follows: $w_i = 1$ means condition c_i is fully important, while $w_i = 0$ means condition c_i is not important at all and can be neglected (and hence should have no impact on the result). Criteria with intermediate weights should still be taken into account, but to a lesser extent than criteria with weight $w_i = 1$. In order to have an appropriate scaling, it is assumed that $\max_i w_i = 1$ [7]. To reflect the impact of a weight on the evaluation of a criterion, a premodification is performed on the initial BSDs, taking into account the weights. Afterwards, the modified BSDs are aggregated using the regular aggregation techniques, as if they were regular, non-modified BSDs.

Modelling the impact of weights on an individual BSD

Suppose that all individual query conditions c_i have been evaluated, resulting each in a BSD $\tilde{b_i}$. When aggregating the individual BSDs, to calculate the global satisfaction degree (also expressed by a BSD) for the entire query, the respective weights have to be taken into account. Although other interpretations of weights are possible, in the presented approach, the weights are considered to have an impact on the individual BSDs. Therefore, before aggregating the individual BSDs, the impact of the weights on these BSDs needs to be calculated first. Let g be the operator that represents this weight influence on the individual BSDs:

$$g : [0, 1] \times \tilde{\mathbb{B}} \to \tilde{\mathbb{B}} : (w_i, \tilde{b_i}) \mapsto g(w_i, \tilde{b_i}) \tag{7}$$

Additionally, let g_s and g_d represent the functions that model the impact of a weight on the satisfaction degree and dissatisfaction degree respectively:

$$g_s \quad : \quad [0, 1] \times [0, 1] \to [0, 1] : (w_i, s_{\tilde{b_i}}) \mapsto g_s(w_i, s_{\tilde{b_i}}) \tag{8}$$

$$g_d \quad : \quad [0, 1] \times [0, 1] \to [0, 1] : (w_i, d_{\tilde{b_i}}) \mapsto g_d(w_i, d_{\tilde{b_i}}) \tag{9}$$

In order to be a suitable operator, g,g_s and g_d need to meet the following requirements [4, 7]:

- for a weight 1, the BSD must remain unchanged, i.e., $g(1, \tilde{b_i}) = \tilde{b_i}$
- because criteria with weight 0 should have no impact on the result, the BSD needs to be mapped to the neutral element for the aggregation. This is (1,0) in case of conjunction, and (0,1) in case of disjunction, i.e.,

  $$g(0, \tilde{b_i}) = (1, 0) \text{ in case of conjunction}$$

  $$g(0, \tilde{b_i}) = (0, 1) \text{ in case of disjunction}$$

- the operators g_s and g_d need to be monotonic in the (dis)satisfaction degree (with $g_\bullet$ either g_s or g_d and x_1,x_2 values for s or d respectively):

$$\forall w, x_1, x_2 \in [0,1] : x_1 \geq x_2 \Rightarrow g_\bullet(w, x_1) \geq g_\bullet(w, x_2)$$

- the operators g_s and g_d need to be monotonic in the weight (with $g_\bullet$ either g_s or g_d and x the value for s or d respectively):

$$\forall w_1, w_2, x \in [0,1] : w_1 \geq w_2 \Rightarrow g_\bullet(w_1, x) \geq g_\bullet(w_2, x)$$

  or

$$\forall w_1, w_2, x \in [0,1] : w_1 \geq w_2 \Rightarrow g_\bullet(w_1, x) \leq g_\bullet(w_2, x)$$

  depending on the kind of the aggregation and the kind of operator (for either s or d)

Remark that because of the difference in neutral element for conjunction and disjunction, the weight impact operator g will behave differently according to the type of aggregation. In conjunctions, weights smaller than 1 will result in BSDs being drawn 'upwards' to values being "more satisfactory" (toward the neutral element (1,0), representing totally satisfied), while in disjunctions weights smaller than 1 will result in BSDs being drawn 'downwards' to values being "more dissatisfactory" (toward the neutral element (0,1), representing totally dissatisfied).

Implication functions f_{im} and coimplication functions f_{im}^{co} can be used to model the impact of weights (this idea, although stated in a different context, has already been introduced in [4],[11]). f_{im} and f_{im}^{co} are [0,1]-valued extensions of Boolean implication and coimplication functions, and hence can be rewritten as $f_{im}(x,y) = \neg x \vee y$ and $f_{im}^{co}(x,y) = \neg f_{im}(\neg x, \neg y) = \neg(\neg(\neg x) \vee \neg y) = (\neg x \wedge y)$. When looking at the extreme points of the first argument $x = 0$ and $x = 1$, $f_{im}(x,y)$ and $f_{im}^{co}(x,y)$ reduce to the following, for all y:

$$\begin{array}{ll} f_{im}(0,y) = 1 & \quad f_{im}(1,y) = y \\ f_{im}^{co}(0,y) = y & \quad f_{im}^{co}(1,y) = 0 \end{array}$$

So, the implication is neutral in the second argument for $x = 1$ and can thus be used with the weight w as first argument and the value of the satisfaction or dissatisfaction degree as the second argument (for weight $w = 1$, the membership grade should remain unchanged). Moreover, for $x = 0$, the result of the implication is drawn toward 1. This makes the implication, with the weight as first argument and the satisfaction or dissatisfaction degree as second argument, a good choice in cases where the degree should be drawn upwards toward 1 for weights $w < 1$. This is the case for the satisfaction degree when working with a conjunction and for the dissatisfaction degree when working with a disjunction.

The coimplication on the other hand is neutral in the second argument for $x = 0$, and can thus be used with $1 - w$ as first argument and the satisfaction or dissatisfaction degree as second argument (for weight $w = 1$, or thus $x = 0$, the membership grade should remain unchanged). Moreover, for $w = 0$ (or $x = 1$), the result of the implication is drawn toward 0. This makes the coimplication, with $1 - w$ as first argument and the satisfaction or dissatisfaction degree as second argument, a good choice in cases where the degree should be drawn downwards toward 0 for weights $w < 1$ ($x > 0$). This is the case for the dissatisfaction degree when working with a conjunction and for the satisfaction degree when working with a disjunction. Based on these observations, the impact of a weight on a BSD can then be defined as follows:

- Weight impact operator for conjunction

$$g^{\wedge} : [0,1] \times \tilde{\mathbb{B}} \to \tilde{\mathbb{B}} : (w, \tilde{b}) \mapsto g^{\wedge}(w, \tilde{b}) = \left(s_{g^{\wedge}(w,\tilde{b})}, d_{g^{\wedge}(w,\tilde{b})}\right) \qquad (10)$$

  where:

$$s_{g^{\wedge}(w,\tilde{b})} = f_{im}(w, s_{\tilde{b}})$$
$$d_{g^{\wedge}(w,\tilde{b})} = f_{im}^{co}(1 - w, d_{\tilde{b}})$$

- Weight impact operator for disjunction

$$g^{\vee} : [0,1] \times \tilde{\mathbb{B}} \to \tilde{\mathbb{B}} : (w, \tilde{b}) \mapsto g^{\vee}(w, \tilde{b}) = \left(s_{g^{\vee}(w,\tilde{b})}, d_{g^{\vee}(w,\tilde{b})}\right) \qquad (11)$$

  where:

$$s_{g^{\vee}(w,\tilde{b})} = f_{im}^{co}(1 - w, s_{\tilde{b}})$$
$$d_{g^{\vee}(w,\tilde{b})} = f_{im}(w, d_{\tilde{b}})$$

Some interesting implication and coimplication functions are:

- The Kleene-Dienes implication and coimplication:

$$\begin{aligned} f_{im_{KD}}(x, y) &= \max(1 - x, y) \\ f_{im_{KD}}^{co}(x, y) &= \min(1 - x, y) \end{aligned} \qquad (12)$$

- The Reichenbach implicator implication and coimplication:

$$\begin{aligned} f_{im_{Rb}}(x,y) &= 1 - x + x \cdot y \\ f^{co}_{im_{Rb}}(x,y) &= (1-x) \cdot y \end{aligned} \tag{13}$$

- The Gödel implication and coimplication:

$$\begin{aligned} f_{im_{Go}}(x,y) &= \begin{cases} 1 \text{ if } x \leq y \\ y \text{ otherwise} \end{cases} \\ f^{co}_{im_{Go}}(x,y) &= \begin{cases} 0 \text{ if } x \geq y \\ y \text{ otherwise} \end{cases} \end{aligned} \tag{14}$$

As an example consider the weight operator for conjunction based on the Kleene-Dienes implicator:

$$g^{\wedge}(w,\tilde{b}) = \big(\max(1-w, s_{\tilde{b}}), \min(w, d_{\tilde{b}})\big) \tag{15}$$

It is easy to see that this is indeed a monotonic operator, where the result for $w = 1$ will reduce to $(s_{\tilde{b}}, d_{\tilde{b}})$ and for $w = 0$ to $(1,0)$, as was required for a suitable conjunction weight operator.

Modelling weighted aggregation

Using the definitions of the weight impact operators $g^{\wedge}$ and $g^{\vee}$, and the basic aggregation operators for BSDs, a definition of an extended operator for weighted conjunction $\tilde{\wedge}^w$ and disjunction $\tilde{\vee}^w$ of BSDs can now be defined:

$$\tilde{\wedge}^w : ([0,1] \times \tilde{\mathbb{B}})^2 \to \tilde{\mathbb{B}} \tag{16}$$

$$((w_1,\tilde{b}_1),(w_2,\tilde{b}_2)) \mapsto g^{\wedge}(w_1,\tilde{b}_1)\tilde{\wedge}g^{\wedge}(w_2,\tilde{b}_2)$$

$$\tilde{\vee}^w : ([0,1] \times \tilde{\mathbb{B}})^2 \to \tilde{\mathbb{B}} \tag{17}$$

$$((w_1,\tilde{b}_1),(w_2,\tilde{b}_2)) \mapsto g^{\vee}(w_1,\tilde{b}_1)\tilde{\vee}g^{\vee}(w_2,\tilde{b}_2)$$

4.2 Averaging

Besides the basic aggregation operators based on t-norm and t-conorms, satisfaction degrees can also be aggregated using other operators, like e.g. averaging operators. The same holds for bipolar satisfaction degrees. Some averaging operators that could be used are the arithmetic mean (AM), geometric mean (GM) or harmonic mean (HM):

$$AM(x_1, , x_n) = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{18}$$

$$GM(x_1, , x_n) = \sqrt[n]{\prod_{i=1}^{n} x_i} \tag{19}$$

$$HM(x_1, , x_n) = \frac{n}{\sum_{i=1}^{n} 1/x_i} \tag{20}$$

These averaging operators cannot be applied on BSDs as such, because a BSD consists of a pair of values. Extended versions of these operators can be defined though, where the above, regular, averaging operators are applied once for all the satisfaction degrees together, and, separately, once for all the dissatisfaction degrees together:

$$\widetilde{AM}(\tilde{b}_1, , \tilde{b}_n) = \left(\frac{1}{n}\sum_{i=1}^{n} s_{\tilde{b}_i}, \frac{1}{n}\sum_{i=1}^{n} d_{\tilde{b}_i}\right) \tag{21}$$

$$\widetilde{GM}(\tilde{b}_1, , \tilde{b}_n) = \left(\sqrt[n]{\prod_{i=1}^{n} s_{\tilde{b}_i}}, \sqrt[n]{\prod_{i=1}^{n} d_{\tilde{b}_i}}\right) \tag{22}$$

$$\widetilde{HM}(\tilde{b}_1, , \tilde{b}_n) = \left(\frac{n}{\sum_{i=1}^{n} 1/s_{\tilde{b}_i}}, \frac{n}{\sum_{i=1}^{n} 1/d_{\tilde{b}_i}}\right) \tag{23}$$

Weights can be applied in two ways to these extended averaging operators. On the one hand, the weights can be statically connected to the query criteria itself, and handled as in the previous subsection. So in that case, the importance of the query criteria is known in advance. On the other hand, weights can also dynamically be applied based on the ranking of how well the query criteria are satisfied, as is the case for Ordered Weighted Averaging (OWA) operators [17, 18] for regular satisfaction degrees. In that case, the final importance of a query criterion is not known in advance but depends on its ranking, based on its satisfaction and dissatisfaction degree, compared to other query criteria.

4.2.1 Weighted Averaging

When statically taking weights into account for the extended averaging operators for BSDs, the weighted counterparts of the traditional averaging operators (e.g. weighted arithmetic mean ($\widetilde{AM}^w$), weighted geometric mean ($\widetilde{GM}^w$) or weighted harmonic mean ($\widetilde{HM}^w$)) can be used for aggregating the satisfaction degrees on the one hand, and, separately, the dissatisfaction degrees on the other hand. In this case, the weights can be associated as described in Subsection 4.1. So weights w_i can be connected to the individual conditions c_i, with $w_i \in [0,1]$. Again, in order to have an appropriate scaling, it is assumed that $\max_i w_i = 1$. As such, extended versions of these weighted averaging operators have been defined as follows:

$$\widetilde{AM}^w : ([0,1] \times \tilde{\mathbb{B}})^n \to \tilde{\mathbb{B}} \tag{24}$$

$$\widetilde{AM}^w((w_1, \tilde{b}_1), \ldots, (w_n, \tilde{b}_n)) = \left(\frac{\sum_{i=1}^n w_i \cdot s_{\tilde{b}_i}}{\sum_{i=1}^n w_i}, \frac{\sum_{i=1}^n w_i \cdot d_{\tilde{b}_i}}{\sum_{i=1}^n w_i} \right)$$

$$\widetilde{GM}^w : ([0,1] \times \tilde{\mathbb{B}})^n \to \tilde{\mathbb{B}} \tag{25}$$

$$\widetilde{GM}^w((w_1, \tilde{b}_1), \ldots, (w_n, \tilde{b}_n)) = \left(\left(\prod_{i=1}^n s_{\tilde{b}_i}^{w_i} \right)^{1/\sum_{i=1}^n w_i}, \left(\prod_{i=1}^n d_{\tilde{b}_i}^{w_i} \right)^{1/\sum_{i=1}^n w_i} \right)$$

$$\widetilde{HM}^w : ([0,1] \times \tilde{\mathbb{B}})^n \to \tilde{\mathbb{B}} \tag{26}$$

$$\widetilde{HM}^w((w_1, \tilde{b}_1), \ldots, (w_n, \tilde{b}_n)) = \left(\frac{\sum_{i=1}^n w_i}{\sum_{i=1}^n \frac{w_i}{s_{\tilde{b}_i}}}, \frac{\sum_{i=1}^n w_i}{\sum_{i=1}^n \frac{w_i}{s_{\tilde{b}_i}}} \right)$$

4.2.2 Ordered Weighted Averaging

Ordered weighted averaging of BSDs can be based on the traditional OWA operators [17, 18] as done in the case of aggregating regular satisfaction degrees. In that case, the weights are no longer attached to the query criteria themselves. The weight vector is still given in advance. But it will be determined dynamically which query criterion will receive which weight, according to the ranking of the resulting satisfaction degrees. Moreover, the normalization constraint on the weights w_i is no longer $\max_i w_i = 1$, but $\sum_{i=1}^n w_i = 1$. The OWA operator for n values $x_1, \ldots, x_n$ is defined by:

$$OWA(x_1, \ldots, x_n) = \sum_{i=1}^n w_i \cdot x'_i \tag{27}$$

with $x'_1, \ldots, x'_n$ the ranked version of $x_1, \ldots, x_n$, i.e., x'_i is the i^{th} largest value of $x_1, \ldots, x_n$. This OWA operator can also be extended to work with BSDs. But, in contrast to the weighted averaging approach above, we cannot apply this traditional OWA operator to the satisfaction degrees and dissatisfaction degrees separately. The reason for this is that the rankings for the satisfaction degrees and the dissatisfaction degrees do not necessarily have to be the same (mostly not), even if for the dissatisfaction degree the i^{th} smallest value would be taken instead of the i^{th} largest value. Therefore, in order to use an extended OWA operator for BSDs, the BSDs need to be considered as a whole, and rank ordered as such, for example by using the ranking function presented in Section 2. The extended OWA operator then becomes (with W_n the set of all possible weight vectors of size n, where $\sum_{i=1}^{n} w_i = 1$):

$$\widetilde{OWA} : W_n \times \mathbb{B}^n \to \mathbb{B} \tag{28}$$

$$\widetilde{OWA}\left((w_1, \ldots, w_n), (\tilde{b}_n, \ldots, \tilde{b}_n)\right) = \left(\sum_{i=1}^{n} w_i \cdot s'_i, \sum_{i=1}^{n} w_i \cdot d'_i\right)$$

Hereby (s'_i, d'_i) are the satisfaction and dissatisfaction degree of the BSD $\tilde{b'}_i$, the i^{th} largest BSD of $\tilde{b}_1, \ldots, \tilde{b}_n$, according to the ranking function used.

Depending on the weight vector that is used, this extended OWA operator will behave differently (just like the regular OWA operator). As special cases, it can, e.g., act as a maximum function for BSDs ($w_1 = 1, w_i = 0$ for $i > 1$), a minimum function for BSDs ($w_n = 1, w_i = 0$ for $i < n$) and a median function for BSDs (for odd n: $w_{\lceil \frac{n}{2} \rceil} = 1, w_i = 0$ for $i \neq \lceil \frac{n}{2} \rceil$, where $\lceil\ \rceil$ denotes the ceiling function; for even n: $w_{\frac{n}{2}} = \frac{1}{2}, w_{\frac{n}{2}+1} = \frac{1}{2}, w_i = 0$ for $i \neq \frac{n}{2}$ and $i \neq \frac{n}{2} + 1$).
Remark that the exact behaviour of the maximum, minimum and median function for BSDs (and also for all other OWA operators) depends on the specific ranking function that is used.

5 Advanced Aggregation

As an example of advanced aggregation of BSDs, this paper will consider the aggregation of BSDs in the context of decision support systems (or multi-criteria decision making systems). Traditional decision support systems do not address the issue of heterogeneous bipolarity. In this section, heterogeneous bipolar criteria satisfaction handling is studied in an evaluation process based on the Logic Scoring of Preference method (LSP method), which is a soft computing method for evaluation and selection of complex systems [10]. As input, a list of attributes a_i, $i = 1, \ldots, n$ is considered for which the decision maker must provide elementary attribute criteria c_i, $i = 1, \ldots, n$ for each component a_i of the array of attributes. In traditional systems, the criteria usually express what is permitted, but sometimes it is easier or even only possible to express what is unpermitted.

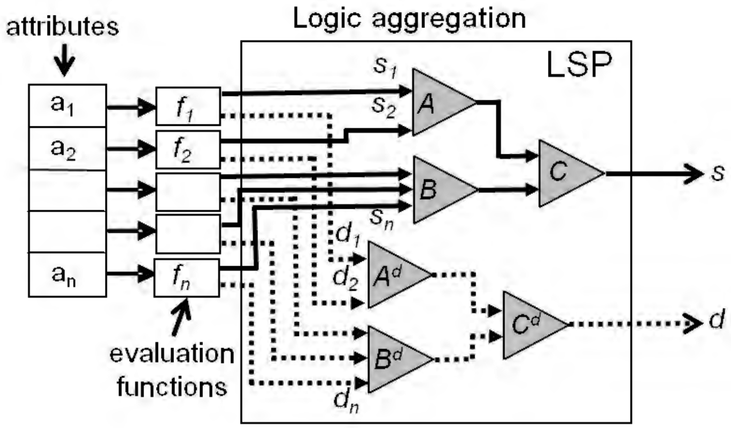

Figure 1: LSP aggregation

Therefor, the presented satisfaction modeling framework based on BSDs will be used here, and the evaluation of each criterion c_i, using evaluation functions f_i, will lead to a BSD (s_i, d_i). Here again s_i denotes the extent of criteria satisfaction and d_i denotes the extent of criteria dissatisfaction.

The BSDs then naturally should also be used to express the global (un)suitability of the different alternatives under consideration. In that case, s denotes to what extent an alternative is *suitable* with respect to the decision to be made, taking into account all criteria, and d denotes to what extent an alternative is *unsuitable* with respect to the decision to be made, taking into account all criteria. Therefor, to come to an overall suitability for an alternative in the decision support system, all BSDs (s_i, d_i), $i = 1, \ldots, n$ must be aggregated. This can be done by two separate preference aggregation structures that are specifically configured for the decision to be made. One structure is used for the aggregation of the satisfaction degrees (s_i) and consists of aggregrators that are denoted by the symbols A, B, C in Fig. 1. The other structure is used for the aggregation of the dissatisfaction degrees (d_i) and consists of dual aggregators which are denoted by A^d, B^d, C^d in the figure. This approach guarantees the independence of satisfaction and dissatisfaction degrees.

Each aggregation structure can include a variety of aggregators and reflects the conditions of the decision logic. All simple aggregators are modelled using

Table 1: Simple aggregator operators in LSP

Operator		**Symbol**	r
Full disjunction		D	$+\infty$
Partial disjunction	Very Strong	D++	20.63
	Strong	D+	9.521
	Medium	DA	3.929
	Weak	D-	2.018
	Very Weak	D- -	1.449
Neutrality		A	1
Partial conjunction	Very Weak	C- -	0.619
	Weak	C-	0.261
	Medium	CA	-0.72
	Strong	C+	-3.510
	Very Strong	C++	-9.06
Full conjunction		C	$-\infty$

the basic generalized conjunction/disjunction function [11]

$$\begin{aligned} GCD &: [0,1]^k \times [0,1]^k \to [0,1] \\ (w_1,\dots,w_k),(e_1,\dots,e_k) &\mapsto (w_1 e_1^r + \cdots + w_k e_k^r)^{1/r} \end{aligned} \tag{29}$$

where k denotes the number of inputs, e_i, $i = 1,\dots,k$ are the satisfaction (or dissatisfaction) degrees of the BSDs involved in the aggregation and w_i, $i = 1,\dots,k$ denotes the adjustable weights that are used to represent the relative importance of the attribute within the aggregation and have to sum up to 1 for each aggregator. The exponent r is the parameter that fully determines the logic properties of the aggregator. Examples of relevant values for r together with their semantics and symbols are given in Table 1.

More complex, compound aggregators, as described in [10] can also be used.

The operators CA, C+, C++ and C are suited to model mandatory criteria and reflect requirements for simultaneous satisfaction of the criteria. If they are not satisfied then the overall suitability is considered unacceptable and rated zero. The operators D, D++, D+, DA, D-, D- -, A, C- - and C- can be used to model nonmandatory criteria. If a nonmandatory criteria is not satisfied that will not cause rejection.

The partial conjunction operators can be used to model asymmetrical partial absorption that aggregates mandatory and desired inputs. If the desired input is 0, this causes a penalty P (the average decrement percentage of the output value),

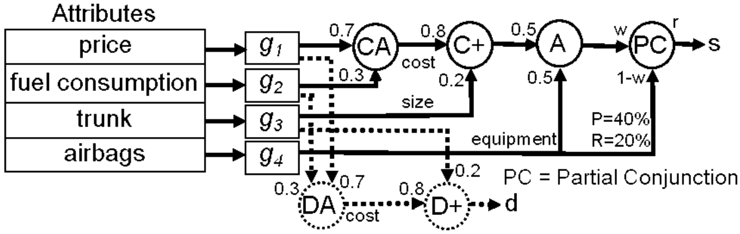

Figure 2: Aggregation of BSDs.

and if the desired input is 1, this causes a reward R (the average increment percentage of the output value). In the case of asymmetrical partial absorption, the parameters of the aggregator (the weights and the andness of partial conjunction) can be computed from the desired penalty/reward pairs. Therefore, the evaluator only has to select the most appropriate percentages of penalty and reward. In the case of other aggregators, the evaluator selects weights that express the desired relative importance of inputs and the andness/orness that reflects the desired level of simultaneity or replaceability of inputs.

Mandatory and nonmandatory criteria are examples of logic conditions that are present in all areas of evaluation. Additional logic conditions include the adjustable levels of simultaneity (andness) and replaceability (orness) that some groups of criteria must satisfy. Finally, all criteria are assumed to have adjustable levels of relative importance. This is the reason why it is convenient to realize the evaluation model using the LSP method.

Example 1 *In Fig. 2, an example of two aggregation schemes is given for a possible decision support system to find the most suited vehicle for a company (which, e.g., is mostly interested in a large car for a small price). The schema for the computation of the overall satisfaction degree s is depicted in straight lines. Partial conjunction CA is used to compute the overall 'cost' satisfaction (based on price and fuel consumption). The 'cost' satisfaction degree is aggregated with the 'size' satisfaction degree (e.g., trunk size) using strong partial conjunction C+. With these two criteria it is reflected that 'cost' and 'size' are mandatory criteria. The equipment criterion (e.g., number of airbags) is desired. Asymmetrical partial absorption, modeled by the aggregation operator A and a partial conjunction operator PC, is used to aggregate the mandatory and desired inputs [10]. The weight w and exponent r of PC are computed from the given penalty P and reward R.*

The schema for the computation of the overall dissatisfaction degree d *is depicted in dotted lines. Only the 'cost' and 'size' criteria are taken into account here. For their aggregation the dual counterpart of their aggregation operators for the satisfaction degree is used. This reflects the dual nature of satisfaction and dissatisfaction.*

It is important to note that the aggregation structure of positive attributes is predominantly conjunctively polarized because evaluators usually simultaneously request all convenient properties. As opposed to that, the aggregation structure of negative attributes is predominantly disjunctively polarized because the evaluated system may be unsuitable if any of the negative attributes is significantly present.

6 Ranking BSDs

After aggregating BSDs (either by regular, weighted or advanced aggregation), a list of results, each with a calculated global BSD, is obtained. In case of 'fuzzy' querying, this list is a list of records, where the BSDs expresses 'query satisfaction'. In case of decision support systems, this list is a list of possible alternatives, where the BSDs express the (un)suitablility of the alternatives. To find the best record or best alternative among the list of possible results, a ranking function for BSDs is required. Again, different ranking functions can be used. One of them, which gives equal importance to the satisfaction degree and the dissatisfaction degree is given in Section 2, i.e.,

$$r_1 = s - d \ \in [-1, 1]. \tag{30}$$

In some cases, it can be useful to give more importance to either the satisfaction degree s or the dissatisfaction degree d. One option is to rank primarily on the satisfaction (respectively dissatisfaction) degree and use the other one (dissatisfaction respectively satisfaction degree) only as a tiebreaker. Another option is to use both degrees for the calculation of the ranking, but to favour one over the other. E.g., a possible ranking function which gives more importance to the satisfaction degree s is

$$r_2 = \frac{s}{s+d} \ \in [0, 1] \tag{31}$$

If $s = 0$, this will always have as resulting ranking $r_2 = 0$. Consequently, when using this ranking function, the positive query conditions will be regarded as strict conditions. If $s = 1$, r_2 will take values in $[0.5, 1]$, which implies that the dissatisfaction degree still has some impact on the ranking.

Another ranking function, assigning more importance to the dissatisfaction degree d, is:

$$r_3 = \frac{1-d}{(1-s)+(1-d)} \in [0,1] \tag{32}$$

If $d = 1$, this will always lead to $r_3 = 0$, while if $d = 0$, r_3 will take values in $[0.5, 1]$, depending on the satisfaction degree.

7 Conclusions

In this paper, an overview of different kinds of aggregation operators for aggregating bipolar satisfaction degrees is given. Basic aggregation is based on the aggregation of Atanassov Intuitionistic Fuzzy Sets. When taking into account possible weights that can indicate a difference in importance for the individual BSDs, one approach uses the standard, non-weighted, aggregation operators for BSDs as underlying operators and uses implication and coimplication functions to model the impact of the weight, prior to the actual aggregation. In that case, distinction has to be made between the weight impact in case of conjunction and in case of disjunction. Another approach for taking into account weights does not use the standard aggregation operators for BSDs, but is based on traditional averaging operators and the weighted version of them. In that case, distinction can be made between averaging operators where the weights are statically connected to the BSDs themselves and averaging operators where the weights are dynamically associated with the criteria. Finally, a more advanced form of aggregation, based on LSP aggregators has been presented. Up to now, LSP aggregators have been mostly used in the context of multi-criteria decision making systems.

References

[1] Atanassov, K. (1986) Intuitionistic fuzzy sets. Fuzzy Sets and Systems, 20, 87–96.

[2] Bosc, P., Pivert, O. (1995) SQLf: A relational database language for fuzzy querying. IEEE Transactions on Fuzzy Systems, 3, 1–17.

[3] Bosc, P., Kraft, D.H., Petry, F.E. (2005) Fuzzy sets in database and information systems: Status and opportunities. Fuzzy Sets and Systems, 156-3, 418–426.

[4] De Tré, G., De Baets, B.: Aggregating Constraint Satisfaction Degrees Expressed by Possibilistic Truth Values. IEEE Transactions on Fuzzy Systems, 11-3, 361–368.

[5] De Tré, G., De Caluwe, R., Kacprzyk, J., Zadrożny, S. (2005) On flexible querying via extensions to fuzzy sets. In: Proc. of the 4th Conf. of the European Society for Fuzzy Logic and Technology (Eusflat'05) and 11th Rencontres Phrancophones sur la logique floue et ses applications (LFA'05), Barcelona, Spain, 1225–1230.

[6] De Tré, G., Zadrożny, S., Matthé, T., Kacprzyk, J., Bronselaer, A. (2009). Dealing with Positive and Negative Query Criteria in Fuzzy Database Querying: Bipolar Satisfaction Degrees. In: Proc. of the 8th International Conference on Flexible Querying and Answering Systems (FQAS 2009), Roskilde, Denmark. Lecture Notes in Artificial Intelligence, 5822, 593–604.

[7] Dubois, D., Fargier, H., Prade, H. (1997) Beyond min aggregation in multicriteria decision: (ordered) weighted min, discri-min and leximin. In: Yager, R.R., Kacprzyk, J. (eds.) The ordered weighted averaging operators: Theory and applications, Kluwer Academic Publishers, Boston, USA, 181–192.

[8] Dubois, D., Prade, H. (2002) Bipolarity in flexible querying. Lecture Notes in Artificial Intelligence, 2522, 174–182.

[9] Dubois, D., Prade, H. (2008) Handling bipolar queries in Fuzzy Information Processing. In: Galindo, J. (ed.) Handbook of Research on Fuzzy Information Processing in Databases, IGI Global, New York, USA, 97–114.

[10] Dujmović, J. (2007) Preference logic for system evaluation. IEEE Transactions on Fuzzy Systems, 15-6, 1082–1099.

[11] Dujmović, J. (2008) Characteristic forms of generalized conjunction/disjunction. In: Proc. of the IEEE World Congress on Computational Intelligence (Fuzz-IEEE 2008), Hong Kong, 1075–1080.

[12] Kacprzyk, J., Zadrożny, S., Ziółkowski, A. (1989) FQUERY III+: a "human-consistent" database querying system based on fuzzy logic with linguistic quantifiers. Information Systems, 14, 443–453.

[13] Lacroix, M., Lavency, P. (1987) Preferences: Putting more knowledge into queries. In: Proc. of the VLDB Conf. (VLDB'87), Brighton, UK, 217–225.

[14] Matthé, T., De Tré, G., Hallez, A. (2009) Impact of Weights on Conjunctive and Disjunctive Aggregation of Extended Possibilistic Truth Values. Fuzzy Sets and Systems, 160-15, 2141–2158.

[15] Matthé, T., De Tré, G. (2009) Bipolar query satisfaction using satisfaction and dissatisfaction degrees: bipolar satisfaction degrees. In: Proc. of the ACM Symposium on Applied Computing (ACM SAC'09), Honolulu, Hawaii, 1699–1703.

[16] Tahani, V. (1977) A conceptual framework for fuzzy query processing: a step toward very intelligent database systems. Information Processing and Management, 13, 289–303.

[17] Yager, R.R. (1988) On ordered weighted averaging aggregation operators in multi-criteria decision making. IEEE Transactions on Systems, Man and Cybernetics, 18, 183–190.

[18] Yager, R. R., Kacprzyk, J. (1997) The Ordered Weighted Averaging Operators: Theory and Applications. Kluwer, Norwell, Massachusetts, USA.

[19] Zadrożny, S., Kacprzyk, J. (2006) Bipolar queries and queries with preferences. In: Proc. of the DEXA Conference (DEXA'06), Krakow, Poland, 415-419.

[20] Zadrożny, S., De Tré, G., De Caluwe, R., Kacprzyk, J. (2008) An Overview of Fuzzy Approaches to Flexible Database Querying. In: Galindo, J. (ed.) Handbook of Research on Fuzzy Information Processing in Databases, IGI Global, New York, USA, 34–54.

The papers presented in this Volume 2 constitute a collection of contributions, both of a foundational and applied type, by both well-known experts and young researchers in various fields of broadly perceived intelligent systems.
It may be viewed as a result of fruitful discussions held during the Tenth International Workshop on Intuitionistic Fuzzy Sets and Generalized Nets (IWIFSGN-2011) organized in Warsaw on September 30, 2011 by the Systems Research Institute, Polish Academy of Sciences, in Warsaw, Poland, Institute of Biophysics and Biomedical Engineering, Bulgarian Academy of Sciences in Sofia, Bulgaria, and WIT - Warsaw School of Information Technology in Warsaw, Poland, and co-organized by: the Matej Bel University, Banska Bystrica, Slovakia, Universidad Publica de Navarra, Pamplona, Spain, Universidade de Tras-Os-Montes e Alto Douro, Vila Real, Portugal, and the University of Westminster, Harrow, UK:

Http://www.ibspan.waw.pl/ifs2011

The consecutive International Workshops on Intuitionistic Fuzzy Sets and Generalized Nets (IWIFSGNs) have been meant to provide a forum for the presentation of new results and for scientific discussion on new developments in foundations and applications of intuitionistic fuzzy sets and generalized nets pioneered by Professor Krassimir T. Atanassov. Other topics related to broadly perceived representation and processing of uncertain and imprecise information and intelligent systems have also been included. The Tenth International Workshop on Intuitionistic Fuzzy Sets and Generalized Nets (IWIFSGN-2011) is a continuation of this undertaking, and provides many new ideas and results in the areas concerned.

We hope that a collection of main contributions presented at the Workshop, completed with many papers by leading experts who have not been able to participate, will provide a source of much needed information on recent trends in the topics considered.

ISBN-13 9788389475411

9 788389 475411