Positive and negative preferences

Stefano Bistarelli[1,2], Maria Silvia Pini[3], Francesca Rossi[3], and K. Brent Venable[3]

[1] University "G' D'Annunzio", Pescara, Italy
bista@sci.unich.it
[2] Istituto di Informatica e Telematica, CNR, Pisa, Italy
Stefano.Bistarelli@iit.cnr.it
[3] University of Padova, Italy
{mpini,frossi,kvenable}@math.unipd.it

Abstract Many real-life problems present both negative and positive preferences. We extend and generalize the existing soft constraints framework to deal with both kinds of preferences. This amounts at adding a new mathematical structure, which has properties different from a semiring, to deal with positive preferences. Compensation between positive and negative preferences is also allowed.

1 Introduction

Many real-life problems present both hard constraints and preferences. Moreover, preferences can be of many kinds:

- qualitative (as in "I like A better than B") or quantitative (as in "I like A at level 10 and B at level 11"),
- conditional (as in "If A happens, then I prefer B to C") or not,
- positive (as in "I like A, and I like B even more than A"), or negative (as in "I don't like A, and I really don't like B").

Our long-term goal is to define a framework where all such kinds of preferences can be naturally modelled and efficiently dealt. In the paper, we focus on problems which present positive and negative, quantitative, and non-conditional preferences.

Positive and negative preferences could be thought as two symmetric concepts, and thus one could think that they can be dealt with via the same operators and with the same properties. However, it is easy to see that this could be not reasonable in many scenarios, since it would not model what usually happens in reality.

For example, when we have a scenario with two objects A and B, if we like both A and B, then the preference of the overall scenario should be even more preferred than both of them. On the other hand, if we don't like A nor B, then the preference of the scenario should be smaller than the preferences of A and B. Thus combination of positive preferences should give us a higher preference, while combination of negative preferences should give us a lower preference.

Also, when having both kinds of preferences, it is natural to have also a element which models "indifference", stating that we express neither a positive nor a negative preference over an object. For example, we may say that we like peaches, we don't like bananas, and we are indifferent to apples. The indifferent element should also behave like the unit element in a usual don't care operator. That is, when combined with any preference (either positive or negative), it should disappear. For example, if we like peaches and we are indifferent to eggs, a meal with peaches and eggs would have overall a positive preference.

Notice that the assumption that composing two good things will give us even better thing, and composing two bad things will give us an even worse thing could be not true in general [7]. So for instance, we may like eating cakes and we may like eating ice-cream, but we don't like to eat them both (too heavy). Or, if we like peaches and we are indifferent to eggs, could be not true that I should like peaches AND eggs (e.g., think of these two sitting on the same plate).

Finally, besides combining positive preferences among themselves, and also negative preferences among themselves, we also have the problem of combining positive with negative preferences. For example, if we have a meal with meat (which we like very much) and wine (which we don't like), then what should be the preference of the meal? To know that, we must combine the positive preference given to meat to the negative preference given to wine.

Soft constraints [3] are a useful formalism to model problems with quantitative preferences. However, they can model just one kind of preferences. In fact, we will see that technically they can model just negative preferences. Informally, the reason for this statement is that preference combination returns lower preferences, as natural when using negative preferences, and the best element in the ordering behaves like indifference (that is, combined with any other element a, it returns a). Thus all the levels of preferences modelled by a semiring are indeed levels of negative preferences.

Our proposal to model both negative and positive preferences consists of the following ingredients:

- We use the usual soft constraint formalism, based on c-semirings, to model negative preferences.
- We define a new structure, with properties different from a c-semiring, to model positive preferences.
- We make the highest negative preference coincide with the lower positive preference; this element models indifference.
- We define a new combination operator between positive and negative preference to model *preference compensation*.

In the framework proposed in [1, 5], positive and negative preferences are dealt with by using possibility theory [4, 10]. This mean that preferences are assimilated to possibilities. In this context, it is reasonable to model the negative preference of an event by looking at the possibility of the complement of such an event. In fact, in the approach of [5], a negative preference for a value or a tuple is translated into a positive preference on the values or tuples different from the one rejected. For example, if we have a variable representing the price of an apartment with domain $\{p_1 = low, p_2 = medium, p_3 = high\}$ then a negative preference stating that a high price (p_3) is rejected with degree 0.9 (almost completely) is translated in giving a positive preference 0.9 to $p_1 \vee p_2$. In our framework, neither positive nor negative preferences are considered as possibilities. Therefore, we do not relate the negative preference of an event to the preference of the complement of such an event.

The paper is organized as follows: Section 2 recalls the main notions of semiring-based soft constraints. Then, Section 3 describes how we model negative preferences using usual soft constraints, Section 4 introduces the new preference structure to model positive preferences, and Section 5 shows how to model both positive and negative preferences. Finally, Section 6 defines constraint problems with both positive and negative preferences, Section 7 summarizes the results of the paper and gives some hints for future work.

2 Background: semiring-based soft constraints

A soft constraint [3] is just a classical constraint where each instantiation of its variables has an associated value from a partially ordered set. Combining constraints will then have to take into account such additional values, and thus the formalism has also to provide suitable operations for combination ($\times$) and comparison ($+$) of tuples of values and constraints. This is why this formalization is based on the concept of semiring, which is just a set plus two operations.

A c-semiring is a tuple $\langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ such that:

- A is a set and $\mathbf{0}, \mathbf{1} \in A$;
- $+$ is commutative, associative, idempotente, $\mathbf{0}$ is its unit element, and $\mathbf{1}$ is its absorbing element;
- $\times$ is associative, commutative, distributes over $+$, $\mathbf{1}$ is its unit element and $\mathbf{0}$ is its absorbing element.

Consider the relation $\leq_S$ over A such that $a \leq_S b$ iff $a + b = b$. Then it is possible to prove that:

- $\leq_S$ is a partial order;
- $+$ and $\times$ are monotone on $\leq_S$;
- $\mathbf{0}$ is its minimum and $\mathbf{1}$ its maximum;
- $\langle A, \leq_S \rangle$ is a lattice and, for all $a, b \in A$, $a + b = lub(a, b)$.

Moreover, if $\times$ is idempotent, then $\langle A, \leq_S \rangle$ is a distributive lattice and $\times$ is its glb.

Informally, the relation $\leq_S$ gives us a way to compare (some of the) tuples of values and constraints. In fact, when we have $a \leq_S b$, we will say that b *is better than* a.

Given a c-semiring $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$, a finite set D (the domain of the variables), and an ordered set of variables V, a constraint is a pair $\langle def, con \rangle$ where $con \subseteq V$ and $def : D^{|con|} \rightarrow A$. Therefore, a constraint specifies a set of variables (the ones in con), and assigns to each tuple of values of D of these variables an element of the semiring set A.

A soft constraint satisfaction problem (SCSP) is a pair $\langle C, con\rangle$ where $con \subseteq V$ and C is a set of constraints over V.

A classical CSP is just an SCSP where the chosen c-semiring is: $S_{CSP} = \langle\{false, true\}, \vee, \wedge, false, true\rangle$. On the other hand, fuzzy CSPs [8, 9] can be modeled in the SCSP framework by choosing the c-semiring: $S_{FCSP} = \langle[0, 1], max, min, 0, 1\rangle$.

For weighted CSPs, the semiring is $S_{WCSP} = \langle\Re^+, min, +, +\infty, 0\rangle$. Preferences are interpreted as costs from 0 to $+\infty$. Costs are combined with $+$ and compared with min. Thus the optimization criterion is to minimize the sum of costs.

For probabilistic CSPs [6], the semiring is $S_{PCSP} = \langle[0, 1], max, \times, 0, 1\rangle$. Preferences are interpreted as probabilities ranging from 0 to 1. As expected, they are combined using $\times$ and compared using max. Thus the aim is to maximize the joint probability.

Given two constraints $c_1 = \langle def_1, con_1\rangle$ and $c_2 = \langle def_2, con_2\rangle$, their *combination* $c_1 \otimes c_2$ is the constraint $\langle def, con\rangle$ defined by $con = con_1 \cup con_2$ and $def(t) = def_1(t \downarrow^{con}_{con_1}) \times def_2(t \downarrow^{con}_{con_2})$[4]. In words, combining two constraints means building a new constraint involving all the variables of the original ones, and which associates to each tuple of domain values for such variables a semiring element which is obtained by multiplying the elements associated by the original constraints to the appropriate subtuples.

Given a constraint $c = \langle def, con\rangle$ and a subset I of V, the *projection* of c over I, written $c \Downarrow_I$, is the constraint $\langle def', con'\rangle$ where $con' = con \cap I$ and $def'(t') = \sum_{t/t\downarrow^{con}_{I\cap con}=t'} def(t)$. Informally, projecting means eliminating some variables. This is done by associating to each tuple over the remaining variables a semiring element which is the sum of the elements associated by the original constraint to all the extensions of this tuple over the eliminated variables.

The *solution* of a SCSP problem $P = \langle C, con\rangle$ is the constraint $Sol(P) = (\bigotimes C) \Downarrow_{con}$. That is, to obtain the solution constraint of an SCSP, we combine all constraints, and then project over the variables in con. In this way we get the constraint over con which is "induced" by the entire SCSP.

Given an SCSP problem P, consider $Sol(P) = \langle def, con\rangle$. A solution of P is a pair $\langle t, v\rangle$ where t is an assignment to all the variables in con and $def(t) = v$.

Given an SCSP problem P, consider $Sol(P) = \langle def, con\rangle$. An optimal solution of P is a pair $\langle t, v\rangle$ such that t is an assignment to all the variables in con, $def(t) = v$, and there is no t', assignment to con, such that $v <_S def(t')$. Therefore optimal solutions are solutions which have the best semiring element among those associated to solutions. The set of optimal solutions of an SCSP P will be written as $Opt(P)$.

Figure 1 shows an example of a fuzzy CSP, two of its solutions one of which (S_2) is optimal.

[4] By $t \downarrow^X_Y$ we mean the subtuple obtained by projecting the tuple t (defined over the set of variables X) over the set of variables $Y \subseteq X$.

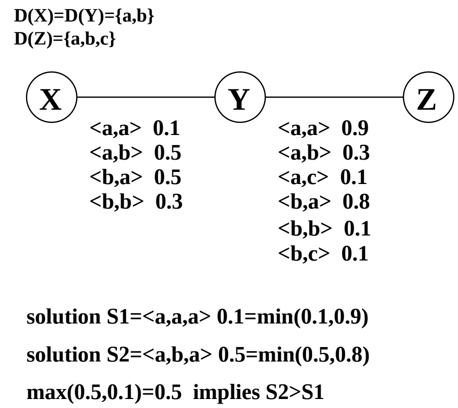

Figure 1. A Fuzzy CSP, two of its solutions, one of which is optimal (S_2).

3 Negative preferences

As anticipated in the introduction, we need two different mathematical structures to deal with positive and negative preferences. For negative preferences, we use the standard c-semiring, while for positive preferences we need to define a new structure. Such two structures are connected by a single element, which belongs to both, and which denotes indifference. Such an element is the best among the negative preferences and the worst one among the positive preferences.

The structure used to model negative preferences is a c-semiring, as defined in Section 2. In fact, in a c-semiring the element which acts as indifference is the $\mathbf{1}$, since $\forall a \in A$, $a \times \mathbf{1} = a$. Element $\mathbf{1}$ is also the best in the ordering, so indifference is the best preference we can express. This means that all the other preferences are less than indifference, thus they are naturally interpreted as negative preferences. Moreover, in a c-semiring combination goes down in the ordering, since $a \times b \leq a, b$. This can be naturally interpreted as the fact that combining negative preferences worsens the overall preference.

This interpretation is very natural when considering, for example, the weighted semiring $(R^+, min, +, +\infty, 0)$. In fact, in this case the real numbers are costs and thus negative preferences. The sum of different costs is worse in general w.r.t. the ordering induced by the additive operator (min) of the semiring.

Let us now consider the fuzzy semiring $([0,1], max, min, 0, 1)$. According to this interpretation, giving a preference equal to 1 to a tuple means that there is nothing negative about such a tuple. Instead, giving a preference strictly less than 1 (e.g., 0.6) means that there is at least a constraint which such tuple doesn't satisfy at the best. Moreover, combining two fuzzy preferences means taking the minimum and thus the worst among them.

When considering classical constraints via the c-semiring $S_{CSP} = \langle \{false, true\}, \vee, \wedge, false, true \rangle$, we just have two elements to model preferences: true and false. True is here the indifference, while false means that we don't like the object. This interpretation is consistent with the fact that, when we don't want to say anything about the relation between two variables, we just omit the constraint, which is equivalent to having a constraint where all instantiations are allowed (thus they are given value true).

In the following of this paper, we will use standard c-semirings to model negative preferences, and we will usually write their elements with a negative index $_n$ and by calling N the carrier set, as follows: $(N, +_n, \times_n, \bot_n, \top_n)$.

4 Positive preferences

As said above, when dealing with positive preferences, we want two main properties: that combination brings to better preferences, and that indifference is lower than all the other preferences. These properties can be found in the following structure, that we will call a positive preference structure.

Definition 1. *A positive preference structure is a tuple* $(P, +_p, \times_p, \bot_p, \top_p)$ *such that*

- P *is a set and* $\top_p, \bot_p \in P$*;*
- $+_p$*, the additive operation, is commutative, associative, idempotent, with* $\bot_p$ *as is its unit element* ($\forall a \in P, a +_p \bot_p = a$) *and* $\top_p$ *as is its absorbing element* ($\forall a \in P, a +_p \top_p = \top_p$)*;*
- $\times_p$*, the multiplicative operation, is associative, commutative and distributes over* $+_p$ ($a \times_p (b +_p c) = a \times_p b +_p a \times_p c$)*,* $\bot_p$ *is its unit element and* $\top_p$ *is its absorbing element.*

Notice that the additive operation of this structure has the same properties as the corresponding one in c-semirings, and thus it induces a partial order over P in the usual way: $a \leq_p b$ iff $a +_p b = b$. Also for positive preferences, we will say that b is better than a iff $a \leq_p b$. As for c-semirings, this allows to prove that $+$ is monotone over $\leq_p$ and it coincides with the least upper bound in the lattice $(P, \leq_p)$.

On the other hand, the multiplicative operation has different properties. More precisely, the best lement in the ordering ($\top_p$) is now the absorbing element, while the worst element ($\bot_p$) is the unit element. This reflects the desired behavior of the combination of positive preferences. In fact, we can prove the following properties.

First, $\times_p$ is monotone over $\leq_p$.

Theorem 1. *Given the positive preference structure* $(P, \times_p, +_p, \bot_p, \top_p)$*, consider the relation* $\leq_p$ *over* P*. Then* $\times_p$ *is monotone over* $\leq_p$*. That is,* $a \times_p d \leq_p b \times_p d$*,* $\forall d \in P$*.*

Proof. Since $a \leq_p b$, by definition, $a +_p b = b$. Thus, $\forall d \in P$ we have that $b \times_p d = (a +_p b) \times_p d$. Since $\times_p$ distributes over $+_p$, $b \times_p d = (a \times_p d) +_p (b \times_p d)$, and thus $a \times_p d \leq b \times_p d$.

Also, combining positive preferences using the multiplicative operator gives an element which is better or equal in the ordering.

Corollary 1. *Given the positive preference structure* $(P, +_p, \times_p, \top_p, \bot_p)$*. For any pair* $a, b \in P$*,* $a \times_p b \geq_p a, b$*.*

Proof. Since $\forall a, b \in P$, $a \geq_p \bot_p$ and $b \geq_p \bot_p$. By monotonicity of $\times_p$ we have $a \times_p b \geq_p \bot_p \times b = b$ and $b \times_p a \geq_p \bot_p \times a = a$.

Notice that this is the opposite behaviour to what happens when combining negative preferences, which brings lower in the ordering.

Since both $+_p$ and $\times_p$ obtain a higher preference, but $+_p$ is the least upper bound, then the following corollary is an obvious consequence.

Corollary 2. *Given the positive preference structure* $(P, +_p, \times_p, \bot_p, \top_p)$*, for any pair* $a, b \in P$*,* $a \times_p b \geq_p a +_p b$*.*

In a positive preference structure, $\bot_p$ is the element modelling indifference. In fact, it is the worst in the ordering and it is the unit element for the combination operator $\times_p$. These are exactly the desired properties for indifference w.r.t. positive preferences.

The role of $\top_p$ is to model a very high preference, much higher than all the others. In fact, since it is the absorbing element of the combination operator, when we combine any positive preference a with $\top_p$, we get $\top_p$ and thus a disappears. A similar interpretation can be given to $\bot_n$ for the negative preferences.

5 Positive and negative preferences

In order to handle both positive and negative preferences we propose to combine the two structures described above as follows.

Definition 2. *A preference structure is a tuple* $(P \cup N, +_p, \times_p, +_n, \times_n, +, \times, \bot, \Box, \top)$ *where*

- $(P, +_p, \times_p, \Box, \top)$ *is a positive preference structure;*
- $(N, +_n, \times_n, \bot, \Box)$ *is a c-semiring;*
- $+ : (P \cup N)^2 \longrightarrow P \cup N$ *is an operator such that* $+_{|N} = +_n$ *and* $+_{|P} = +_p$*, and such that* $a_n + a_p = a_p$ *for any* $a_n \in N$ *and* $a_p \in P$*.*
- $\times : (P \cup N)^2 \longrightarrow P \cup N$ *is an operator such that* $\times_{|N} = \times_n$ *and* $\times_{|P} = \times_p$*, which respects properties P1, P2, and P3 defined later in this section.*

Notice that a partial order on the structure $(P \cup N)$ is defined by saying that $a \leq b \iff a + b = b$. Easily we have $\bot \leq \Box \leq \top$. In details, there is a unique maximum element coinciding with $\top$, a unique minimum element coinciding with $\bot$, and the element $\Box$, which is smaller than any positive preferences and greater than any negative preference, and which is used to model indifference. Such an ordering is shown in Figure 2.

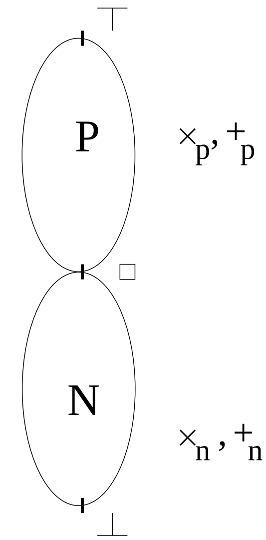

Figure 2. A preference structure.

$\times$ is defined by extending the positive and negative multiplicative operator in order to allow the combination of heterogeneous preferences. Its definition have to take in account the possibility of a compensation between positive and negative preferences. Informally, we will define a way to relate elements of P to elements of N s.t. their combination could compensate and give as a result the indifference element $\Box$. To do that, we

- Partition both P and N in the same number of classes. Each of the class of P (N) contains elements which behave similarly when combined with elements of the "opposite" class in N (P). Such classes will be technically defined by using an equivalence relation among elements with some specific properties.
- Define and ordering among the classes and a correspondence function mapping each class in its opposite. The result are two ordering, one among positive class and the other among negative ones, that are exactly the same w.r.t. the correspondence function.

We consider two equivalence relations $\equiv_p$ and $\equiv_n$ over P and N respectively. For any element a of $P \cup N$ let us denote with $[a]$ the equivalence class to which a belongs. Such equivalence relations must satisfy the following properties:

- $|N/\equiv_n| = |P/\equiv_p|$ (i.e. $\equiv_n$ and $\equiv_p$ have the same number of equivalence classes).
- $[a] \leq_{\equiv} [b]$ iff $\forall x \in [a]$ and $\forall y \in [b]$, $x \leq y$.
- there must exist at least a bijection f such that f: $N/\equiv_n \longrightarrow P/\equiv_p$ and $[a] \leq_{\equiv} [b]$ iff $f([a]) \geq_{\equiv} f([b])$ where $[a]$ and $[b]$ are classes built from negative preferences.

Notice that for the case where the orders on N and P are total it's natural to define the equivalence classes to be intervals, so that $\leq_\equiv$ is also a total order.

The multiplicative operator of the preference structure, written $\times$, must satisfy the following properties:

P1. $a \times b = \Box$ iff $f([b]) = [a]$;
P2. if $[a] \leq_\equiv [b]$ then $\forall c \in P \cup N$, $a \times c \leq b \times c$; that is, $\times$ is monotone w.r.t. the ordering $\leq_\equiv$;
P3. $\times$ is commutative.

Summarizing, to define a preference structure, we need the following ingredients:

- $P, \times_p, +_p$;
- $N, \times_n, +_n$;
- $\top, \bot, \Box$;
- $\times$, defined by giving $\equiv_p, \equiv_n$, and f.

Given these properties, it is easy to show that the combination of a positive and a negative preference is a preference which is higher than, or equal to, the negative one and lower than, or equal to, the positive one.

Theorem 2. *Given a preference structure* $(P, N, +_p, \times_p, +_n, \times_n, +, \times, \bot, \Box, \top)$, *we have that, for any* $p \in P$ *and* $n \in N$, $n \leq p \times n \leq p$.

Proof. By monotonicity of $\times$, and since $n \leq \Box \leq p$ for any $n \in N$ and $p \in P$,, we have the following chain: $n = n \times \Box \leq n \times p \leq \Box \times p = p$.

This means that the compensation of positive and negative preferences must lie in one of the chains between the two given preferences. Notice that all such chains pass through the indifference element $\Box$.

Moreover, we can be more precise: if we combine p and n, and we compare $f([n])$ to $[p]$, we can discover if $p \times n$ is in P or in N, as the following theorem shows.

Theorem 3. *Given a preference structure* $(P, N, +_p, \times_p, +_n, \times_n, +, \times, \bot, \Box, \top)$, *take any* $p \in P$ *and any* $n \in N$. *Then we have:*

- *if* $f([n]) \leq_\equiv [p]$, *then* $\Box \leq_p p \times n \leq_p p$*'*
- *if* $f([n]) >_\equiv [p]$, *then* $n \leq_p p \times n \leq_p \Box$.

Proof. If $f([n]) \leq_\equiv [p]$, then for any element c in $f([n])$, $c \leq_p p$. By motononicity of $\times$, we have $\Box = n \times c \leq_p n \times p$. Similarly for $p \times n \leq_p \Box$ when $f([n]) >_\equiv [p]$.

Notice that the multiplicative operator $\times$ might be not associative. In fact, consider for example the situation with two occurrences of a positive preference p and one negative preference n such that $[p] = f([n])$. That is, p and n compensate completely to indifference. Assume also that $\times_p$ is idempotent. Then, $p \times (p \times n) = p \times \Box = p$, while $(p \times p) \times n = p \times n = \Box$. This depends on the fact that we are free to choose $\times_n$ and $\times_p$ as we want, and $\times$ concides with them when used on preferences of the same kind. Certainly, if any one of $\times_p$ or $\times_n$ is idempotent, then $\times$ is not associative. However, there are also cases in which both $\times_p$ and $\times_n$ are not idempotent, and still $\times$ is not associative. This means that, when combining all the preferences in a problem, we must choose an association ordering.

The preference structure we defined allows us to have different ways to model and reason about positive and negative preferences.In fact, besides the combination operator, which has different properties by definition, we can also have different lattices $(P, \leq_p)$ and $(N, \leq_n)$. This means that we can have, for example, a richer structure for positive preferences w.r.t. the negative ones. This is normal in real-life problems, where not necessarily we want the same expressivity when expressing negative statements and positive ones. For example, we could be satisfied with just two levels of negative preferences, but we might want ten levels of positive preferences. Of course our framework allows us also to model the case in which the two structures are isomorphic.

Notice that classical soft constraints, as anticipated above, refer only to negative preferences in our setting. This means that, by using soft constraints, we can express many levels of negative preference (as many as the elements of the semiring), but only one level of positive preference, which coincide also with the indifference element and also with the top element.

6 Bipolar preference problems

We can extend the notion of soft constraint allowing preference functions to associate to partial instatiations either positive or negative preferences.

Definition 3 (bipolar constraints). *Given a preference structure $(P, N, +_p, \times_p, +_n, \times_n, +, \times, \bot, \Box, \top)$, a finite set D (the domain of the variables), and an ordered set of variables V, a constraint is a pair $\langle def, con \rangle$ where $con \subseteq V$ and $def : D^{|con|} \rightarrow P \cup N$.*

A Bipolar CSP (V, C) is defined as a set of variables V and a set of bipolar constraints C.

A solution of a bipolar CSP can then be defined as follows.

Definition 4 (solution). *A solution of a bipolar CSP (V, C) is a complete assignment to all variables in V, say s, and an associated preference $pref(s) = (p_1 \times_p \ldots \times_p p_k) \times (n_1 \times_n \ldots \times_n n_l)$, where for $i := 1, \ldots, k$ $p_i \in P$ and for $j := 1, \ldots, l$ $n_j \in N$ and $p_i = def_i(s \downarrow^V_{var(c_i)})$ where $var(c_i)$ are the variables involved in the constraint $c_i \in C$.*

In words, the preference of a solution s is obtained by:

1. combining all the positive preferences associated to all its projections using $\times_p$;

2. combining all the negative preferences associated to all its projections using $\times_n$;
3. then, combining the positive preference obtained at steps 1 and the negative preference obtained at step 2 using $\times$.

Notice that this way of computing the preference of a solution is by choosing to combine all the preferences of the same kind together before combining them with preferences of the other kind. Other choices could lead in general to different results due to the possible non-associativity of the $\times$ operator.

Definition 5 (optimal solution). *An optimal solution of a bipolar CSP* (V, C) *is a pair* $\langle s, pref(s)\rangle$ *such that* s *is an assignment to all the variables in* V*, and there is no* s'*, assignment to* V*, such that* $pref(s) < pref(s')$*.*

Therefore optimal solutions are solutions which have the best preference among those associated to solutions. The set of optimal solutions of a bipolar CSP B will be written as $Opt(B)$.

Figure 3 shows a bipolar constraint which associates positive and negative preferences to its tuples. In this example we use the weighted c-semiring $(R^+, min, +, 0, +\infty)$ for representing the negative preferences. For the positive preferences, we consider separately two positive preference structures: $(R^+, max, +, +\infty, 0)$ and $(R^+, max, max, +\infty, 0)$. Notice that the indifference element coincides with 0 in both the positive and negative preference structures. In the example in Figure 3, we assume that every equivalence class is composed by a single preference, and that function f is the identity. Moreover, when applied to one positive and one negative preference, $\times$ is the arithmetic sum of positive/negative numbers denoted as $+^p_n$. Therefore we consider two preference structures: the first one is $(R^+, R^+, max, +, min,\ +, max - min, +^p_n, +\infty,\ 0, +\infty)$, and the second one is $(R^+, R^+, max, max, min,\ +, max - min, +^p_n, +\infty, 0, +\infty)$ where $max - min$ is the $+$ operator of the structure (induced by the max and min operators of the positive and negative preferences rispectively).

In Figure 3, preferences belonging to P have index p, while those belonging to N have index n. The left part of Figure 3 shows the bipolar CSP, while the right part shows the preference associated to each solution. For example, for solution $(x = a, y = b)$, we must combine 1_p, 10_p, and 10_n. To do this, we must compute $(1_p \times_p 10_p)$. If $\times_p = +$, then the result is 11_p. If instead $\times_p = max$, then the result is 10_p. Then, such result must be combined with 10_n, giving in the first case $11_p \times 10_n = 1_p$, and in the second case $10_p \times 10_n = 0$.

7 Future work

We have extended the semiring-based formalisms for soft constraints to be able to handle both positive and negative preferences. We are currently studying which properties are needed in order to obtain completely specular preference structure where the times operator $\times$ satisfy the associativity property.

We are also studying the correlation between our work and the works on non-monotonic concurrent constraints [2]. In this framework the language is

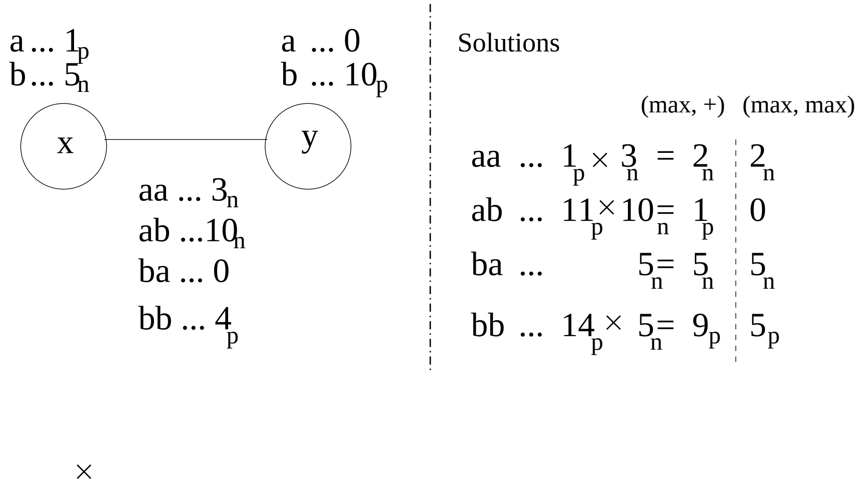

Figure 3. A bipolar CSP with both positive and negative preferences, and its solutions.

enlarged with a *get* operator that remove constraints from the store. It seems that removing a constraint could be equivalent to adding a positive constraint.

Further work will concern the possible use of constraint propagation techniques in this framework, which may need adjustments w.r.t. the classical techniques due to the possible non-associative nature of the compensation operator.

References

1. S. Benferhat, D. Dubois, S. Kaci, H. Prade. Bipolar representation and fusion of preferences in the possibilistic logic framework. Proc. KR 2002, 158-169, 2002.
2. E. Best, F.S. de Boer, C. Palamidessi. Partial Order and SOS Semantics for Linear Constraint Programs. Proc. of Coordination 97, vol. 1282 of LNCS, pages 256-273. Springer-Verlag, 1997.
3. S. Bistarelli, U. Montanari and F. Rossi. Semiring-based Constraint Solving and Optimization. Journal of the ACM, Vol.44, n.2, March 1997.
4. D. Dubois, H. Fargier, H. Prade. Possibility theory in constraint satisfaction problems: handling priority, preference and uncertainty. Applied Intelligence, 6, 287-309, 1996.
5. D. Dubois, S. Kaci, H. Prade. Bipolarity in reasoning and decision - An introduction. The case of the possibility theory framework. IPMU 2004.
6. H. Fargier and J. Lang. Uncertainty in constraint satisfaction problems: a probabilistic approach. In *Proc. European Conference on Symbolic and Qualita tive Approaches to Reasoning and Uncertainty (ECSQARU)*, pages 97–104. Springer-Verlag, LNCS 747, 1993.
7. S.O. Hansson. The Structure of Values and Norms. Cambridge University Press, 2001.
8. Zs. Ruttkay. Fuzzy constraint satisfaction. In *Proc. 3rd IEEE International Conference on Fuzzy Systems*, pages 1263–1268, 1994.

9. T. Schiex, H. Fargier, and G. Verfaille. Valued Constraint Satisfaction Problems: Hard and Easy Problems. In *Proc. IJCAI95*, pages 631–637. Morgan Kaufmann, 1995.
10. L.A. Zadeh. Fuzzy sets as a basis for the theory of possibility. Fuzzy Sets and Systems, 13-28, 1978.