Soft Constraints for Lexicographic Orders*

Fabio Gadducci[1], Mattias Hölzl[2], Giacoma V. Monreale[1], and Martin Wirsing[2]

[1] Dipartimento di Informatica, Università di Pisa, Pisa
[2] Institut für Informatik, LMU, Münich

Abstract. While classical Constraint Satisfaction Problems (CSPs) concern the search for the boolean assignment of a set of variables that has to satisfy some given requirements, their soft variant considers ordered domains for assignments, thus modeling preferences: the aim is to provide an environment where suitable algorithms (e.g. on constraint propagation) can be stated and proved, and inherited by its instances.

Besides their flexibility, these formalisms have been advocated for their modularity: suitable operators can be defined, in order to manipulate such structures and build new ones. However, some intuitive constructions were given less attention, such as lexicographic orders.

Our works explores such orders in three instances of the soft CSP framework. Our results allow for a wider application of the formalism, and it is going to be pivotal for the use of constraints in modeling scenarios where the features to be satisfied are equipped with a fixed order of importance.

Keywords: Soft constraints, lexicographic orders.

1 Introduction

Classical Constraint Satisfaction Problems (CSPs) concern the search for those assignments to a set of variables that may satisfy a family of requirements [17]. In order to decrease the complexity of such a search, various heuristics have been proposed. One of such families is labelled as "constraint propagation": it refers to any reasoning that consists of explicitly forbidding some values or their combinations for the variables of a problem because a subset of its constraints would not be satisfied otherwise. A standard technique for implementing constraint propagation is represented by local consistency algorithms [17, § 3].

The soft framework extends the classical constraint notion in order to model preferences: the aim is to provide a single environment where suitable properties (best known so far are those concerning local consistency [9], and recently also on bucket partitioning [16]) can be proven and inherited by all the instances. Technically, this is done by adding to the classical CSP notion a representation of the levels of satisfiability of each constraint. Albeit appearing with alternative presentations in the literature, the additional component consists of a poset

* Partly supported by the EU FP7-ICT IP ASCEns and by the MIUR PRIN CINA.

F. Castro, A. Gelbukh, and M. González (Eds.): MICAI 2013, Part I, LNAI 8265, pp. 68–79, 2013.
© Springer-Verlag Berlin Heidelberg 2013

(stating the desirability among levels) equipped with a binary operation (defining how two levels can be combined): see e.g [5,19] for two seminal accounts.

The use of these soft formalisms has been advocated for two main reasons. First of all, for their flexibility: their abstract presentation allows for recasting many concrete cases previously investigated in the literature. A bird-view of the expressiveness of the approach can be probably glimpsed at [1]. As important as flexibility is modularity: suitable operators can be defined, in order to manipulate such structures and build new ones that verify the same properties, hence, which are still amenable to the same tools and techniques.

Two classical proposals are based on valuation structures, where the values associated to each constraint are taken respectively from a totally ordered monoid [19] and a c-semiring [5]. Also, a novel and more general formalism based on families of partially ordered monoids was introduced in [13], specifically targeted to tackle operators representing lexicographic orders. The correspondence between the former two proposals was identified early on [2]. The aim of the paper is to recast the latter approach in the standard mold of soft constraints technology. To this end, it provides a novel classification of those three formalisms in terms of ordered monoids, which allows to relate them uniformly. The wealth of techniques introduced in each single approach can then be easily transferred among them. As a main testbed, the paper explores the ability of the three approaches in modeling lexicographic orders.

Our proposal pushes partially ordered valuation structures as a more general proposal for soft constraints, which precisely corresponds to what are called ic-monoids in [13] (§ 2). We first consider a few instances, and a case study illustrating the various constructions (§ 3). We then investigate the definition of operators representing lexicographic orders (§ 4 and § 5), proving some properties for them (§ 6) and applying the formalism to the case study (§ 7). A concluding section recalls the rationale of our work and wraps up the paper (§ 8).

2 Three Domains for Assignments

This section considers three different proposals for the domain of values in the soft constraint framework, and establishes some connections among them. First, however, it recalls some basic mathematical notions.

Definition 1. *A partial order (PO) is a pair $\langle A, \leq \rangle$ such that A is a set of values and $\leq \subseteq A \times A$ is a reflexive, transitive, and anti-symmetric relation. A partial order with top (POT) is a triple $\langle A, \leq, \top \rangle$ such that $\langle A, \leq \rangle$ is a PO, $\top \in A$ and $\forall a \in A. a \leq \top$. A join semi-lattice (JSL) is a POT such that for each finite, not-empty set $X \subseteq A$ there exists a least upper bound (LUB) $\bigvee X$.*

Should we also include the LUB for the empty set, $\bigvee \emptyset$ would coincide with the bottom element $\perp$ of the PO, i.e., $\forall a \in A. \perp \leq a$.

A PO is a total order (TO) if either $a \leq b$ or $b \leq a$ holds for all $a, b \in A$.

Definition 2. *A commutative monoid with identity (CMI) is a triple $\langle A, \otimes, \mathbf{1} \rangle$ such that A is a set of values and $\otimes : A \times A \to A$ is an associative and commutative function with $\mathbf{1}$ as identity, i.e., $\forall a \in A. a \otimes \mathbf{1} = a$.*

An example of CMI is represented by natural numbers with addition $\langle \mathbb{N}, +, 0\rangle$.

2.1 The Domains ...

We open with a novel structure that adapts to the standard soft constraints technology the proposal concerning what are called ic-monoids in [13].

Definition 3. *A* partially-ordered valuation structure *(PVS) is a 4-tuple* $G = \langle A, \leq, \otimes, \top\rangle$ *such that* $G^{\leq} = \langle A, \leq, \top\rangle$ *is a POT,* $G^{\otimes} = \langle A, \otimes, \top\rangle$ *is a CMI, and monotonicity holds.*

A PVS G is bounded *if* $G^{\leq}$ *has bottom element* $\bot$.

Stating that the monoidal operator is monotone means that $a \leq b \Rightarrow a \otimes c \leq b \otimes c$ holds for all $a, b, c \in A$. Also note that if G is bounded then the bottom is the (necessarily unique) absorbing element for $\otimes$, i.e., $a \otimes \bot = \bot$ for all $a \in A$.

We now recall the formalism advanced by [19][1].

Definition 4. *A* valuation structure *(VS) is a PVS G such that* $G^{\leq}$ *is a TO.*

Finally, we recall the definition of absorptive semirings (also c-semirings [5]), recasting it in terms of suitable POs (see also [3]).

Definition 5. *A* semiring valuation structure *(SVS) is a 4-tuple* $G = \langle A, \leq, \otimes, \top\rangle$ *such that* $G^{\leq} = \langle A, \leq, \top\rangle$ *is a JSL,* $G^{\otimes} = \langle A, \otimes, \top\rangle$ *is a CMI, and distributivity holds.*

The monoidal operator $\otimes$ is distributive if $a \otimes \bigvee X = \bigvee\{a \otimes x \mid x \in X\}$ holds for any finite, non-empty subset X of A. Also, recall that if G is bounded, then the LUB exists also for the empty set, and the law trivially holds for that case.

2.2 ... and Their Connections

We now turn our attention to the relationship between the three proposals.

Lemma 1. *Let G be a SVS. Then, it is also a PVS.*

Indeed, from distributivity it easily follows that the tensor operator is monotone, thus implying the lemma above.

In turn, both PVSs and SVSs generalise VSs, replacing their TO with a PO and a JSL, respectively. Moreover, SVSs are well-known as an alternative presentation for c-semirings, as shown e.g. in [3] (where they are referred to with their classical name of absorptive semirings). Indeed, the relationship between VSs and SVSs is even tighter, as stated by the lemma below (see [2, Section 4]).

Lemma 2. *Let G be a VS. Then, it is also a SVS.*

Indeed, in a TO each finite set X admits a maximum, i.e., $\bigvee X \in X$. It is easy to check that this fact and monotonicity imply distributivity.

The relationship among the three domains is summed up in Fig. 1.

[1] The reader should be advised that the chosen order for VSs is the dual of the one adopted in the valuation structure literature. It is clearly an equivalent formalism: our choice allows for a simpler comparison with PVSs and SVSs (see below).

VS (TOT, monotone) [19] $\subseteq$ SVS (JSL, distributive) [5] $\subseteq$ PVS (POT, monotone) [13]

Fig. 1. Relationships among the three domains (for TOT a TO with top)

3 Instances and Examples

This section shows some domains for preferences adopted in the soft constraint literature, as well as some examples of the operators on VSs and SVSs defined there. It is further shown that these domains and operators can be instantiated also for the PVSs formalisms, and later on they are applied in a quite simple case study, which has been adapted from [13].

3.1 Some Instances . . .

The starting point is represented by the results in Section 2, relating VSs and SVSs as instances of the more general notion of PVSs.

A few VSs. We thus first recall some instances of VSs.

The simplest example is the boolean algebra $B = \langle\{\bot, \top\}, \Rightarrow, \vee, \top\rangle$.

Possibly, the most prominent example of a VS is the so-called tropical semiring $T = \langle\mathbb{N}, \geq, +, 0\rangle$, with $\geq$ the inverse of the usual order notion on naturals, thus such that 0 is the top element [15]. It is used for modelling problems where a cost has to be minimised, as in shortest path scenarios under the name of weighted constraint satisfaction [14].

Another variant considers a truncated segment of the natural numbers: $T_n = \langle[0, n], \geq, +_t, 0\rangle$, such that $m +_t o$ is either $m + o$ or, should it be that $m + o \geq n$, just n. Both variants have been extensively studied, since most VSs can be reconnected to these case studies (see e.g. [8]).

Otherwise, the monoidal operator can be replaced by the standard natural multiplication, for $M = \langle\mathbb{N}^+, \geq, \cdot, 1\rangle$, with $\mathbb{N}^+$ the positive natural numbers. The complementary segment $[0, 1]$ over the rational numbers (possibly extending to the real ones) defines the probabilistic VS $F = \langle[0, 1], \leq, \cdot, 1\rangle$ [7].

A few operators for PVSs. The most standard operator for SVSs is the cartesian product: applied to SVSs G_0 and G_1, it returns a SVS whose set of elements is the cartesian product $A_0 \times A_1$, while the other components are obviously defined. As for most operators (a notable exception being the lexicographic order in Section 5), the class of VSs is not closed under it.

As a side remark, it is often the case that it is not possible to decompose an SVS as the product of two or more VSs: as a counterexample, it suffices to consider two VSs G_0 and G_1 and the lifting $(G_0 \times G_1)_\bot$, adding a new bottom element (see also the final remark on Section 6).

Most important is the case of multi-criteria SVSs, as considered in e.g. [4]. Let us start by considering an SVS $G = \langle A, \leq, \otimes, \top \rangle$: let $H(A)$ be the Hoare power-domain, i.e., the set of down-closed (with respect to $\leq$) subsets of A, and $\leq_H$ the order $S_1 \leq_H S_2$ if $S_1 \subseteq S_2$. In other terms, the usual powerset construction for A, up-to removing some of its subsets. Now, for an SVS $G = \langle A, \leq, \otimes, \top \rangle$, its power-domain $H(G)$ is defined as $\langle H(A), \leq_H, \otimes_H, \{\top\} \rangle$, for $\otimes_H$ the obvious extension to sets. The LUB is just set union, and furthermore it is easy to show that distributivity holds, so that $H(G)$ is also an SVS.

In the case study we will consider precisely those two operators we just described. It is relatively simple to show that the class of PVSs is closed under them. Additionally, it behaves better with yet another operator, the lexicographic one, which is also going to be needed for the case study. The main properties for the latter operator will be described in the following sections.

3.2 . . . and a Simple Example

The running example in [13] models a scenario concerning the scheduling of meetings. Among other requirements, each person has to express a possible degree of preference for a given date, and s/he must moreover explicitly state how much her/his presence is actually crucial for that meeting.

We can revise and enrich the original example, using some of the instances and operators just defined. We may assume to have three meetings to organise, each one of them among five possible dates. Some dates might not be compatible, since they overlap (at least, each date overlaps with itself) or because some people are not willing to have too close meetings. For the three meetings, each date (in fact, also each pair and each triple of dates) has associated a set of possible values, representing e.g. the interest and willingness of the persons to be in. The problem is to find the set of three dates maximising such features.

Summing up, each value states how much a person is crucial for a meeting, and her/his willingness to appear in a given date. Also, we may record the status of a person among the group of the possible attendees of the meetings (e.g., the position in a firm or the level of expertise in a technical team).

The last feature can be modelled by T, with 0 the top, and it is an important condition to guarantee. Instead, the statement of the relevance of a person for a meeting can be expressed as B, and its willingness by F.

In order to model the preference domain for this scenario we have to take into account the three features, by attaching more importance to the last two, i.e., the relevance of a person for a meeting and her/his willingness to participate. Indeed, a meeting should be scheduled in the date for which the willingness of relevant people is maximal, while the status of any person should just influence the quality of the best solutions with respect to the other two parameters.

As we will show later on in Section 7, the situation could be modelled by using a lexicographic operator between the two PVSs $B \times F$ and T.

4 Deriving Lexicographic Orders

Building on some classical results for partial orders (see e.g. [11]), we are now going to introduce the key operator for the lexicographic operator of PVSs: it is used in later sections to take two PVSs and to build a new one whose order corresponds to the lexicographic ordering of the two underlying structures.

4.1 Some Facts on Lexicographic Orders for Cartesian Products

This section states some (mostly well-known) properties of lexicographic orders: the characterization of such orders for PVSs is built upon these results.

Definition 6. *Let $\langle A_0, \leq_0 \rangle$ and $\langle A_1, \leq_1 \rangle$ be POs. Then, the associated lexicographic order $\leq_l$ on $A_0 \times A_1$ is given by*

$$\langle a_0, a_1 \rangle \leq_l \langle b_0, b_1 \rangle \text{ if } \begin{cases} a_0 <_0 b_0 & \text{or} \\ a_0 =_0 b_0 \ \& \ a_1 \leq_1 b_1 \end{cases}$$

with $a < b$ meaning that $a \leq b$ and $a \neq b$.

It is easy to see that the order $\leq_l$ is a partial one, and that the occurrence of either top or bottom elements is preserved.

Lemma 3. *Let $P_0 = \langle A_0, \leq_0 \rangle$ and $P_1 = \langle A_1, \leq_1 \rangle$ be POs. Then, also $P_0 \times_l P_1 = \langle A_0 \times A_1, \leq_l \rangle$ is so. Moreover, if both P_0 and P_1 are either POTs or bounded, then also $P_0 \times_l P_1$ is so.*

A tighter property holds if the underlying orders are total.

Lemma 4. *Let $\langle A_0, \leq_0 \rangle$ and $\langle A_1, \leq_1 \rangle$ be TOs. Then, also $\langle A_0 \times A_1, \leq_l \rangle$ is so.*

With some calculations, it can be shown that the lexicographic order forms a JSL if both underlying orders are so, and the latter is bounded.

Proposition 1. *Let $\langle A_0, \leq_0 \rangle$ and $\langle A_1, \leq_1 \rangle$ be JSLs such that $\langle A_1, \leq_1 \rangle$ is bounded. Then, also $\langle A_0 \times A_1, \leq_l \rangle$ is a JSL.*

Proof. The first step is to provide an explicit candidate for the LUB operator. So, let $X \subseteq A_0 \times A_1$ be a finite set, $X_i = \{a_i \mid \langle a_0, a_1 \rangle \in X\}$ the projection of X on the i-th component, and $m(X) = \{a_1 \in X_1 \mid \langle \bigvee X_0, a_1 \rangle \in X\}$. Then

$$\bigvee X = \langle \bigvee X_0, \bigvee m(X) \rangle$$

Note that $m(X)$ might be empty, should $\bigvee X_0 \not\in X_0$, thus the LUB would be $\langle \bigvee X_0, \bot_1 \rangle$, hence the requirement that $\langle A_1, \leq_1 \rangle$ must be bounded.

Now, let us consider an element $\langle c_0, c_1 \rangle$ that is an upper bound of X, i.e., such that $\forall \langle a_0, a_1 \rangle \in X. \langle a_0, a_1 \rangle \leq_l \langle c_0, c_1 \rangle$: this means that $\bigvee X_0 \leq_0 c_0$. The proof then proceeds by case analysis. If $\bigvee X_0 <_0 c_0$, it is straightforward. If instead $\bigvee X_0 = c_0$, then it holds $\bigvee m(X) \leq_1 c_1$: by definition of $\leq_l$, should $\bigvee X_0 \in X_0$, and by the presence of $\bot_1$ in the second component of $\bigvee X$, otherwise.

4.2 Choosing the Right Carrier

Building on this, we can finally consider the structure that we are going to adopt for lexicographic PVSs. We further need an additional definition, in order to manipulate (and restrict) the cartesian product of the carriers of two POs.

Definition 7. *Let $G = \langle A, \leq, \otimes, \top \rangle$ be a PVS. The set $C(A)$ of its* collapsing *elements is defined as $\{c \in A \mid \exists a, b \in A. a < b \wedge a \otimes c = b \otimes c\}$.*

Note that if G is bounded, then clearly $\bot \in C(A)$. In the following, we define A^C as the set of those elements of A that are not collapsing, i.e., $A^C = A \setminus C(A)$.

We can now define the construction we intend to exploit for obtaining PVSs representing lexicographic orders.

Definition 8. *Let $G_0 = \langle A_0, \leq_0, \otimes_0, \top_0 \rangle$ and $G_1 = \langle A_1, \leq_1, \otimes_1, \top_1 \rangle$ be PVSs such that G_1 is bounded. Then, the associated lexicographic carrier is defined as*

$$A_0 \times_l A_1 = (A_0^C \times A_1) \cup (C(A_0) \times \{\bot_1\})$$

In other terms, we restrict our attention to those pairs whose first components is not collapsing, with the exception of those whose second component is the bottom (thus including the bottom element $\langle \bot_0, \bot_1 \rangle$, should G_0 be bounded).

Given the results in the previous section for lexicographic orders over cartesian products, equivalent properties can be stated also for the smaller carrier.

Proposition 2. *Let $G_0 = \langle A_0, \leq_0, \otimes_0, \top_0 \rangle$ and $G_1 = \langle A_1, \leq_1, \otimes_1, \top_1 \rangle$ be PVSs such that G_1 is bounded and both $\langle A_0, \leq_0 \rangle$ and $\langle A_1, \leq_1 \rangle$ are POs (bounded POs, POTs, TOs, JSLs). Then, also $\langle A_0 \times_l A_1, \times_l \rangle$ is so.*

Proof. All proofs are straightforward, except the one for JSLs. The candidate LUB for a finite set $X \subseteq A_0 \times_l A_1$ is still the same as in the proof of Proposition 1. Should $\bigvee X_0$ be not collapsing, or $m(X)$ empty, obviously $\langle \bigvee X_0, \bigvee m(X) \rangle \in A_0 \times_l A_1$. Otherwise, let $\bigvee X_0 \in X_0$ be collapsing. In this case, though, it must be that $m(X) = \{\bot_1\}$, thus the result holds.

We close this section by establishing some closure properties for the set A^C that are going to be needed later on.

Lemma 5. *Let $\langle A, \leq, \otimes, \top \rangle$ be a PVS. Then, $a \otimes b \in C(A)$ iff $\{a, b\} \cap C(A) \neq \emptyset$.*

In other terms, for any PVS both sets $C(A)$ and A^C are closed under the monoidal operator. Moreover, if $a \otimes b \in A^C$, then $\{a, b\} \subseteq A^C$.

5 Deriving Valuation Structures

Building on the characterization results stated in the previous section, we are now ready to move to define the lexicographic operators for PVSs.

First of all, we recall that given two monoids $\langle A_0, \otimes_0, \top_0 \rangle$ and $\langle A_1, \otimes_1, \top_1 \rangle$, the cartesian product of their carriers $A_0 \times A_1$ can obviously be equipped with a monoidal tensor $\otimes_p$ that is defined point-wise, with $\langle \top_0, \top_1 \rangle$ as the identity.

Definition 9. *$G_0 = \langle A_0, \leq_0, \otimes_0, \top_0 \rangle$ and $G_1 = \langle A_1, \leq_1, \otimes_1, \top_1 \rangle$ be PVSs such that G_1 is bounded. Then, the lexicographic structure is defined as*

$$Lex(G_0, G_1) = \langle A_0 \times_l A_1, \leq_l, \otimes_p, \langle \top_0, \top_1 \rangle \rangle$$

Recall that $A_0 \times_l A_1$ discards all the elements $\langle a_0, a_1 \rangle$ of the cartesian product $A_0 \times A_1$ such that a_0 is collapsing and $a_1 \neq \bot_1$. The intuition for it is rather straightforward. Should any collapsing element appear in the first component, there would no chance for $\otimes_p$ to be monotone, unless, of course, the second is the absorbing element $\bot_1$.

Proposition 3. *Let G_0 and G_1 be PVSs such that G_1 is bounded. Then, also $Lex(G_0, G_1)$ is a PVS.*

Proof. The crucial observation is that the triple $\langle A_0 \times_l A_1, \otimes_p, \langle \top_0, \top_1 \rangle \rangle$ is a monoid, i.e., that it is closed under the monoidal operator $\otimes_p$. Indeed, the only problem may occur for those pairs $\langle a_0, a_1 \rangle$ and $\langle b_0, b_1 \rangle$ such that either a_0 or b_0 belongs to $C(A_0)$. Suppose that $a_0 \in C(A_0)$ and $b_0 \in A_0^C$. By Lemma 5, also $a_0 \otimes_0 b_0 \in C(A_0)$, so $a_1 \otimes_1 b_1$ must be equal to $\bot_1$. It is indeed so: since $a_0 \in C(A_0)$, then $a_1 = \bot_1$ and so is $\bot_1 \otimes b_1$, hence no problem arises.

As noted in the previous section, the lexicographic order associated to two total orders is also total. This property also holds for VSs.

Proposition 4. *Let G_0 and G_1 be VSs such that G_1 is bounded. Then, also $Lex(G_0, G_1)$ is a VS.*

The theorem does not hold for SVSs since in general distributivity fails. We are however able to state a weaker result, which we consider an interesting contribution to the literature on semiring-based formalisms.

Theorem 1. *Let G_0 be a VS and G_1 a bounded SVS. Then, also $Lex(G_0, G_1)$ is an SVS.*

Proof. As it often occurs in these situations, the only property that is difficult to check is distributivity: explicitly, taking into account the definition of the monoidal operator $\otimes_p$, it must be shown that for any finite, non-empty subset $X \subseteq A_0 \times_l A_1$ the equality below holds

$$\langle a_0, a_1 \rangle \otimes_p \bigvee_l X = \bigvee_l \{ \langle a_0 \otimes_0 b_0, a_1 \otimes_1 b_1 \rangle \mid \langle b_0, b_1 \rangle \in X \}$$

Now, since G_0 is a VS, for any finite, non-empty set X_0 we have $\bigvee X_0 \in X_0$ and moreover, since $\otimes_0$ is also monotone, $a_0 \otimes_0 \bigvee X_0 = \bigvee \{ a_0 \otimes_0 b_0 \mid b_0 \in X_0 \}$.

Thus, should $\bigvee X_0$ be collapsing, also $a_0 \otimes_0 \bigvee X_0$ is so, thus $\bigvee m(X) = \bot_1$ and the equality immediately holds.

If instead $\bigvee X_0$ is not collapsing, recall that either a_0 is collapsing and $a_1 = \bot_1$ or a_0 is not collapsing. In the first case, also $a_0 \otimes_0 \bigvee X_0$ is collapsing, and the equality holds, since the second component is always $\bot_1$. In the second case, $a_0 \otimes_0 \bigvee X_0$ is not collapsing and the equality holds by distributivity of $\otimes_1$.

6 Some Alternative Properties

In this section we are going to investigate some properties of the structure of the set of collapsing elements of a PVS.

On (weakly) strict PVSs. We start considering a counterpart property with respect to the set of collapsing elements.

Definition 10. *Let $G = \langle A, \leq, \otimes, \top \rangle$ be a PVS. We say that it is* strict *if $C(A) = \emptyset$; it is* weakly strict *if it is bounded and $C(A) = \{\bot\}$.*

Thus, if a PVS G_0 is strict, then $A_0 \times_l A_1$ turns out to coincide with $A_0 \times A_1$, and no pair has to be discarded. Should instead G_0 be weakly strict, we have that $A_0 \times_l A_1$ is $((A_0 \setminus \{\bot_0\}) \times A_1) \cup \{\langle \bot_0, \bot_1 \rangle\}$. Indeed, many PVS instances are going to fall into this situation.

Note how strictness can be rephrased in terms of a standard notion concerning the monoidal operator. We say that a monoidal operator $\otimes$ is strictly monotone if $a < b$ implies $a \otimes c < b \otimes c$ for all elements $a, b, c \in A$; it is weakly so if the property holds whenever $c \neq \bot$.

Lemma 6. *Let $G = \langle A, \leq, \otimes, \top \rangle$ be a PVS. It is strict if the monoidal operator $\otimes$ is strictly monotone. It is weakly strict if it is bounded and the monoidal operator $\otimes$ is weakly strict monotone.*

On cancellative PVSs. In the soft constraint literature, weakly strict VSs are called strictly monotonic VSs [7]. An alternative characterisation for such structures that is simpler to verify can be found below. As before, we say that a monoidal operator $\otimes$ is cancellative if $a \otimes c = b \otimes c$ implies $a = b$ for all elements $a, b, c \in A$; it is weakly so if the property holds whenever $c \neq \bot$.

Definition 11. *Let $G = \langle A, \leq, \otimes, \top \rangle$ be a PVS. We say that it is* cancellative *if the monoidal operator $\otimes$ is so. It is* weakly cancellative *if it is bounded and the monoidal operator $\otimes$ is weakly so.*

Note that any (weakly) cancellative PVS is (weakly) strict. The vice versa does not hold in general, except for VSs (see [3]).

Lemma 7. *Let G be a VS. If it is (weakly) strict, then it is also (weakly) cancellative.*

It follows by the fact that in a VS we have $a \vee b \in \{a, b\}$ for any element $a, b \in A$: thus, if $a \otimes c = b \otimes c$ it must be that $a = b$, since the alternative $a < b$ (or its symmetric choice) would imply that also $a \otimes c \neq b \otimes c$.

On idempotent monoids. The situation is quite less convenient for those PVSs such that the monoidal operator $\otimes$ is idempotent, i.e., such that $a \otimes a = a$ for all $a \in A$. In this case $C(A) = A$, since $a \otimes b$ is the greatest lower bound of a and b (see [5]). Thus, if G_0 is idempotent we have that $A_0 \times_l A_1$ coincides with $A_0 \times \{\bot_1\}$, hence with A_0.

On adding bounds to monoids. Finally, let us consider structures possibly lacking the bottom element. Such a requirement has to be enforced in some situations, e.g. when requiring a PVS to be strict. However, removing and later adding a bottom element is never a problem. Indeed, it is easy to show that for any PVS G a new one $G_\perp$ can be obtained, such that $A_\perp = A \uplus \{\perp\}$ and furthermore $\perp \leq_\perp a$ and $a \otimes_\perp \perp = \perp$ for all $a \in A$.

There are two relevant properties of the lifting. First of all, it adds no identification among the elements of the original PVS: $a \leq b$ iff $a \leq_\perp b$ for all $a, b \in A$. Hence, any PVS can become bounded without any loss in the precision of the order on the preferences. Related to this fact, the lifting also inherits the relevant properties we discussed in this section so that if G is either strict or cancellative, then it is also weakly so.

7 Applying the Lexicographic Operator

The scheduling example introduced in Section 3.2 shows the usefulness of the lexicographic operator. We can indeed use this operator to model the preference domain for the scenario we described.

As said previously, we have to take into account three features: for each person the relevance for a meeting and the willingness, which can be modelled by $B \times F$, ans well as the status, which can be modelled by T.

As observed in Section 3.1, T is a VS. Differently from T_n, it is cancellative, so the good properties discussed in the previous section hold. In particular, it can be always used as the first component of any lexicographic construction. However, in order to use it as the second component, as it is necessary in the scheduling example, its bounded version T_∞ has to be considered, obtained by adding the infinite ∞ as the bottom element.

We also know that the cartesian product applied to SVSs B and F returns a new SVS $B \times F$, therefore we can model the preference domain for the scheduling scenario as $Lex(B \times F, T_\infty)$. Clearly, the unique collapsing element of SVS $B \times F$ is the bottom $\langle\perp, 0\rangle$, hence, since T_∞ is a bounded VS, then $Lex(B \times F, T_\infty)$ is a PVS, with carrier $((\{\perp, \top\} \times [0, 1]) \setminus \{\langle\perp, 0\rangle\}) \times T_\infty$ plus the bottom $\langle\perp, 0, \infty\rangle$.

In this case, note that preference values $\langle\langle\top, 0\rangle, 0\rangle$ and $\langle\langle\perp, 1\rangle, 0\rangle$ are not comparable, since the first component is so: $\top$ dominates $\perp$ but 0 does not dominate 1. As it can occur with POs, no value is better than the other: the LUB of these two elements is indeed different from both of them. However, usually one would like to retain the information on all these alternative solutions, and not the LUB of these which could correspond to no real solution. This is the classical situation of multi-criteria constraint satisfaction: we can then exploit the Hoare power-domain (see section 3.1), thus modeling the preference domain of our example as $H(Lex(B \times F, T_\infty))$.

8 Conclusions and Future Works

In this paper we presented three possible preference domains to be adopted in the soft constraint techniques. Two of them (VSs and SVSs) were well-known, while the other one (PVSs) has been adapted to the standard soft constraint mold from the ordered monoids proposal in [13]. We studied their connections, and we investigated which of them is best suited for capturing the lexicographic operator. Finally, we discussed some instances of VSs, and a few operators for SVSs, relating them to PVSs, and rounding up the section with a simple example to illustrate the use of the lexicographic order.

Each one of those approaches has its own strength. The complexity of local consistency is actually one of the key issues in the work of VSs, as an intuitive generalization of weighted CSPs [6]. What this paper has shown is that PVSs are the most flexible proposal, as far as modularity is concerned. SVSs instead represent an intermediate stage, as far as expressiveness and (possibly) efficiency is concerned, yet they still allows for suitable constraint propagation techniques whose feasibility has still to be checked for PVSs.

Indeed, the introduction of PVSs might be the starting point on a reflection about the overall properties that are required for the domain of preferences in a soft constraint satisfaction problem. We recall that the soft framework has been introduced in order to provide a single environment where suitable properties (e.g. on constraint propagation or on branch&bound resolution techniques) of constraint satisfaction problems can be proved once and for all the instances. In [13], it has been shown that standard branch&bound techniques can be applied to constraint satisfaction problems whose preference domain is a PVS. When originally introduced, both VSs and SVSs were defined for idempotent monoidal operators, and this allows for the application of a simple constraint propagation algorithm based on arc consistency [12,18]. Later, both structures were extended with a notion of *residuation*, assuming the existence of a partial inverse operator (whenever $a \leq b$, then an element $a \ominus b$ is defined, satisfying $(a \ominus b) \otimes b = a$), such that a more complex variant of the same propagation algorithm can be applied [10,3,9]. It has to be proved that the same algorithm can be used for idempotent or (suitable variants of) residuated PVSs, even if these structures lack the distributivity law.

References

1. Bistarelli, S.: Semirings for Soft Constraint Solving and Programming. LNCS, vol. 2962. Springer, Heidelberg (2004)
2. Bistarelli, S., Fargier, H., Montanari, U., Rossi, F., Schiex, T., Verfaillie, G.: Semiring-based CSPs and Valued CSPs: Frameworks, properties, and comparison. Constraints 4(3), 199–240 (1999)
3. Bistarelli, S., Gadducci, F.: Enhancing constraints manipulation in semiring-based formalisms. In: Brewka, G., Coradeschi, S., Perini, A., Traverso, P. (eds.) ECAI 2006. FAIA, vol. 141, pp. 63–67. IOS Press, Amsterdam (2006)
4. Bistarelli, S., Gadducci, F., Larrosa, J., Rollon, E., Santini, F.: Local arc consistency for non-invertible semirings, with an application to multi-objective optimization. Expert Systems with Applications 39(2), 1708–1717 (2012)
5. Bistarelli, S., Montanari, U., Rossi, F.: Semiring-based constraint solving and optimization. Journal of ACM 44(2), 201–236 (1997)

6. Cohen, D.A., Cooper, M.C., Jeavons, P., Krokhin, A.A.: The complexity of soft constraint satisfaction. Artificial Intelligence 170(11), 983–1016 (2006)
7. Cooper, M.C.: Reduction operations in fuzzy or valued constraint satisfaction. Fuzzy Sets and Systems 134(3), 311–342 (2003)
8. Cooper, M.C.: High-order consistency in valued constraint satisfaction. Constraints 10(3), 283–305 (2005)
9. Cooper, M.C., de Givry, S., Sanchez, M., Schiex, T., Zytnicki, M., Werner, T.: Soft arc consistency revisited. Artificial Intelligence 174(7-8), 449–478 (2010)
10. Cooper, M.C., Schiex, T.: Arc consistency for soft constraints. Artificial Intelligence 154(1-2), 199–227 (2004)
11. Geil, O., Pellikaan, R.: On the structure of order domains. Finite Fields and Their Applications 8(3), 369–396 (2002)
12. Van Hentenryck, P., Deville, Y., Teng, C.-M.: A generic arc-consistency algorithm and its specializations. Artificial Intelligence 57(2-3), 291–321 (1992)
13. Hölzl, M.M., Meier, M., Wirsing, M.: Which soft constraints do you prefer? In: Rosu, G. (ed.) WRLA 2008. ENTCS, vol. 283(2), pp. 189–205. Elsevier, Amsterdam (2009)
14. Larrosa, J., Schiex, T.: In the quest of the best form of local consistency for Weighted CSP. In: Gottlob, G., Walsh, T. (eds.) IJCAI 2003, pp. 239–244. Morgan Kaufmann, San Francisco (2003)
15. Pin, J.-E.: Tropical semirings. In: Gunawardena, J. (ed.) Idempotency, pp. 50–69. Cambridge University Press, Cambridge (1998)
16. Rollon, E., Larrosa, J., Dechter, R.: Semiring-based mini-bucket partitioning schemes. In: IJCAI 2013. Morgan Kaufmann, San Francisc (2013)
17. Rossi, F., van Beek, P., Walsh, T.: Handbook of Constraint Programming. Foundations of Artificial Intelligence. Elsevier, Amsterdam (2006)
18. Schiex, T.: Arc consistency for soft constraints. In: Dechter, R. (ed.) CP 2000. LNCS, vol. 1894, pp. 411–424. Springer, Heidelberg (2000)
19. Schiex, T., Fargier, H., Verfaillie, G.: Valued constraint satisfaction problems: Hard and easy problems. In: IJCAI 1995, pp. 631–637. Morgan Kaufmann, San Francisco (1995)