Enabling Local Computation for Partially Ordered Preferences

Hélène Fargier[1], Emma Rollon[2] and Nic Wilson[3]

[1]IRIT, 118 route de Narbonne 3162 Toulouse Cedex, France
[2]Technical University of Catalonia, Jordi Girona 1-3 08034, Barcelona, Spain
[3]Cork Constraint Computation Centre, University College Cork, Ireland
fargier@irit.fr, erollon@lsi.upc.edu, n.wilson@4c.ucc.ie

Abstract. Many computational problems linked to uncertainty and preference management can be expressed in terms of computing the marginal(s) of a combination of a collection of valuation functions. Shenoy and Shafer showed how such a computation can be performed using a local computation scheme. A major strength of this work is that it is based on an algebraic description: what is proved is the correctness of the local computation algorithm under a few axioms on the algebraic structure. The instantiations of the framework in practice make use of totally ordered scales. The present paper focuses on the use of partially ordered scales and examines how such scales can be cast in the Shafer-Shenoy framework and thus benefit from local computation algorithms. It also provides several examples of such scales, thus showing that each of the algebraic structures explored here is of interest.

Keywords: Soft CSP; Dynamic programming; Valuation networks/algebra.

1 Introduction

Many computational problems linked to reasoning under uncertainty can be expressed in terms of computing the marginal(s) of the combination of a collection of (local) valuation functions. Shenoy and Shafer [23, 22] showed how such a computation can be performed using only local computation (see also, in particular, [14]). A major strength of this work, is that it is based on an algebraic description: what is proved is the correctness of the local computation algorithm under a few axioms on the algebraic structure. Hence, the same algorithm may be used in computing the projection on a given variable of a joint probability distribution described by a Bayesian network, in making the fusion of several basic probability assignments with Dempster's rule of combination, or in computing the degree of consistency of a possibilistic knowledge base.

One important class of problems encompassed by the Shafer-Shenoy framework is optimization. In these problems, solutions are given a score and ranked according to a given order, and one has to find the best solutions according to that order. When the order is total, there exists one optimal score. However,

Article published in Constraints Volume 15, Number 4, 516-539,
DOI: 10.1007/s10601-010-9094-z
http://www.springerlink.com/content/alt370515m33510v/
The final publication is available at www.springerlink.com

when the order is partial, there exists a family of alternatives the score of which are incomparable—hence a set of optimal scores.

In practice, applications of Shenoy and Shafer's framework rely on totally ordered scales of scoring, like the MAX CSP problem [10] or the more general VCSP problem [21]. On the other hand, AI has witnessed the emergence of many frameworks for reasoning based on partial orders. Let us for instance cite semiring-based constraint problems [2, 3], reasoning with preferences [4, 25, 12], reasoning under uncertainty [7, 24], belief revision [16, 1] and default reasoning [5, 11].

The purpose of this paper is to show whether and how optimization problems based on partial orders can be cast in the Shafer-Shenoy framework, so as to provide them with local computation algorithms. To that end, we introduce a general algebraic structure called a *preference degree structure*, and show that different optimization problems, based either on total or partial orders, can be captured by this structure. Then, we propose two different transformations, called *extension* of the order and *set encoding*, that ensure the optimality of the solutions computed using a local computation algorithm. The first one generates only one optimal score, while the second generates the whole set of optimal scores.

The structure of the paper is as follows. Section 2 reviews the Shafer-Shenoy axiomatic framework. Section 3 defines the preference degree structures and provides examples of problems captured by these algebraic structures in order to show that they are of interest. Section 4 presents our main contribution. It shows when and how optimization problems based on partial orders and described under preference degree structures can be cast as instances of the Shafer-Shenoy framework and thus can benefit from the local computation machinery. Section 5 concludes and points out directions of future work. Proofs of the results are included in the appendix.

This paper is an extended version of [9].

2 Axioms for local computation

We recall here some basics of the Shafer-Shenoy abstract axiomatic framework [23, 22, 14] for local computation of multivariate problems. The framework concerns valuations—each of which represents information regarding a set of variables—and two operations: combination and marginalization. Many computational problems can be expressed as marginalizing the combination of a collection of valuations. Such a global computation can often be performed in a more efficient way, using only local operations, if the operations satisfy certain axioms.

Consider a finite set $X = \{x_1, \ldots, x_n\}$ of variables, each x_i ranging over a finite state space (or "domain") D_i. For $S \subseteq X$, D_S will denote the Cartesian product of the domains of variables in S. The elements of D_S are called *tuples*. When S is the empty set , we adopt the convention that $D_\emptyset$ consists of a single object $\Diamond$, so that $D_\emptyset = \{\Diamond\}$. For $d \in D_S$, the *projection/restriction* of d to a subset of variables $Y \subseteq S$, denoted $d[Y]$, is a subtuple of d containing only the elements corresponding to the set Y. Given a function f over D_S, when d is defined over a superset of S, we sometimes write $f(d)$ as an abbreviation for $f(d[S])$.

In the Shafer-Shenoy framework, there is a set V_S associated with each set of variables $S \subseteq X$. The elements of V_S are called *valuations* and S is the *scope* of each $\sigma \in V_S$ — we write $scope(\sigma) = S$. Intuitively, σ represents some information about the set of tuples in D_S. $\mathcal{V} = \bigcup_{S \subseteq X} V_S$ is the set of all valuations. Valuations are primitives in the Shafer-Shenoy framework and as such require no definition. They are simply entities that can be *combined* and *marginalized*:

– The *combination* of two valuations σ and τ, denoted $\sigma \boxtimes \tau$, is a new valuation whose scope is $scope(\sigma) \cup scope(\tau)$.
– The *marginalization* of one valuation σ over a set of variables $T \subseteq scope(\sigma)$, denoted $\sigma^{\downarrow T}$, is a new valuation whose scope is T.

A *valuation system* is a triplet $(\mathcal{V}, \boxtimes, \downarrow)$. A *valuation network* (VN) is a finite set (or multiset) $\Sigma = \{\tau_1, \dots, \tau_m\} \subseteq \mathcal{V}$. The task of interest over Σ is to compute its marginal over a subset T of the set of variables involved:

$$(\boxtimes \Sigma)^{\downarrow T} = (\tau_1 \boxtimes \dots \boxtimes \tau_m)^{\downarrow T}$$

We assume that computing a combination is time and space exponential in the cardinality of the union of the scopes of the valuations involved; this assumption is valid for the instances of the framework we consider in this paper, as well as most other instances. A brute-force computation of the above task will first compute the combination of all valuations, and then its marginalization over T, and so it is exponential in the cardinality of the union of scopes of all valuations, and thus impractical. However, if the valuation system satisfies the Shafer-Shenoy axioms, the marginalization task can be done by sequential marginalization over subsets of valuations and, as a consequence, potentially much more efficiently. This is called *local computation*. We write the Shafer-Shenoy axioms as follows:

Axiom A1: If $S \subseteq T \subseteq scope(\sigma)$, then $(\sigma^{\downarrow T})^{\downarrow S} = \sigma^{\downarrow S}$.

Axiom A2: $\boxtimes$ is associative and commutative.

Axiom A3: $\downarrow$ distributes over $\boxtimes$. Namely, if $scope(\sigma) \subseteq T \subseteq scope(\sigma) \cup scope(\tau)$, then $(\sigma \boxtimes \tau)^{\downarrow T} = \sigma \boxtimes \tau^{\downarrow T \cap scope(\tau)}$.

As shown in [23, 22, 14], if the valuation system Σ satisfies Axioms A1, A2, A3, then $\Sigma^{\downarrow T}$ can be computed by successively eliminating variables in $Y = X \setminus T$ from Σ, where X is the set of variables involved in Σ. For $y_i \in Y$, we define

$$Elim_{y_i}(\Sigma) = \Sigma_{\neg y_i} \cup \{(\boxtimes \Sigma_{y_i})^{\downarrow X \setminus \{y_i\}}\}$$

where $\Sigma_{\neg y_i} = \{\sigma \in \Sigma : y_i \notin \mathit{scope}(\sigma)\}$ is the subset of valuations in Σ whose scope does not contain/*bear on* variable y_i and $\Sigma_{y_i} = \Sigma \setminus \Sigma_{\neg y_i}$ is the subset of valuations that do. It can be proved that

$$(\boxtimes\Sigma)^{\downarrow X \setminus \{y_i\}} = \boxtimes \mathit{Elim}_{y_i}(\Sigma)$$

So, we can go from Σ to a new set of valuations, not bearing on y_i, by combining all the valuations that bear on y_i, computing its marginal over $X \setminus \{y_i\}$ and adding it to the set of valuations that do not bear on $\{y_i\}$. Applying this principle iteratively with respect to all variables in Y, the algorithm computes the marginal of Σ over $T = X \setminus Y$ as:

$$(\boxtimes\Sigma)^{\downarrow X \setminus \{y_1,\ldots,y_k\}} = \boxtimes \mathit{Elim}_{y_k}(\mathit{Elim}_{y_{k-1}}(\ldots \mathit{Elim}_{y_1}(\Sigma)))$$

Axioms for local computation are sufficient conditions for the correctness of the sequential elimination procedure. They also ensure the correctness of algorithms of message passing in a join tree decomposition of the valuation network. The time and space complexity of these algorithms is exponential in a structural parameter called *tree-width* (and thus lower than the time and space complexities of the brute force algorithm). What is important for the purpose of the present paper, is that it is granted that when axioms A1, A2 and A3 hold such algorithms are available.

3 Optimization in preference degree structures

Combinatorial optimization problems can be defined in terms of a set of *variables* taking values on finite sets of *domain values* and a set of *local functions* defined over these variables. Roughly speaking, solving a problem is somehow related to assigning domain values to the variables, evaluating the functions on those assignments and keeping the *best* scores/assignments according to a particular preference order. Different attempts have been made to capture different kinds of optimization problems within a common formal framework. We propose a general framework based on an algebraic structure called *preference degree structure*. This framework captures different optimization problems both based on total and partial orders, as we show in the collection of examples provided.

3.1 Preference degree structures

Let L be a set of values and $\preceq$ be a binary relation that compares the values. We adopt the convention that $\forall a, b \in L, a \preceq b$ means that score a is better than score b, i.e., we are oriented toward minimization. We use notation $\prec$ for the associated strict relation (i.e., $\forall a, b \in L,\ a \prec b$ iff $a \preceq b$ and not$(b \preceq a)$) and $\sim$ for the corresponding indifference relation. Let $\otimes$ be a binary operation: $\otimes : L \times L \to L$. Then,

Definition 1 *A preference degree structure is a triplet $\langle L, \preceq, \otimes \rangle$ which forms an ordered commutative monoid. Its neutral element will be denoted* $\mathbf{1}$.

That is to say, $\preceq$ is a partial order: a reflexive, anti-symmetric and transitive relation over L (hence $a \sim b$ iff $a = b$), and L is equipped with an internal operation $\otimes$ which is associative, commutative and monotonic with respect to $\preceq$ ($a \preceq b \implies a \otimes c \preceq b \otimes c$) and such that $a \otimes \mathbf{1} = a$ for all a.

If there exists an associative and commutative (and idempotent) operator $\oplus$ such that $a \preceq b \iff a \oplus b = a$, then we say that $\oplus$ *represents* $\preceq$. It is well known that such an operation $\oplus$ exists if and only if any pair of elements of L have a greatest lower bound. Notice also that if $\preceq$ is a total order this operator necessarily exists ($\oplus = \min$).

3.2 Preference networks

Combinatorial optimization problems can be defined in terms of a preference degree structure.

Definition 2 *Given a preference degree structure $\langle L, \preceq, \otimes \rangle$ and a set of variables X, a* local function *c is defined to be a function from D_Y to L, for some $Y \subseteq X$. We define scope$(c) = Y$. A* preference network *$\mathcal{C}$ is a set (or multiset) of local functions $\{c_1, \ldots, c_m\}$.*

Each tuple $d \in D_X$ receives a collection $\langle c_1(d), ..., c_m(d) \rangle$ of scores. The global score of d is the aggregation of all the $c_i(d)$ according to $\otimes$.

Definition 3 *Given a preference network $\mathcal{C}$ on $\langle L, \preceq, \otimes \rangle$, the global score of $d \in D_X$ is $score_{\mathcal{C}}(d) = \bigotimes_{c_i \in \mathcal{C}} c_i(d)$. We shall also write $Scores(\mathcal{C}) = \{score_{\mathcal{C}}(d) : d \in D_X\}$.*

The task is generally to compute all or one of the *best* global scores associated to tuples in D_X according to $\preceq$. Formally,

Definition 4 *Given a preference network $\mathcal{C}$ on $\langle L, \preceq, \otimes \rangle$, $d \in D_X$ is an* optimal solution *if there is no $d' \in D_X$ such that $score_{\mathcal{C}}(d') \prec score_{\mathcal{C}}(d)$. A value $a \in L$ is an* optimal score *for $\mathcal{C}$ if $a = score_{\mathcal{C}}(d)$ for some optimal solution d.*

Let us denote $Kernel_{\preceq}(A)$ (the kernel of A) as the set of $\preceq$-minimal elements of any set A (i.e., the set of elements $a \in A$ such that there exists no $b \in A$ with $b \prec a$). It is easy to see that:

Proposition 1 *Given a preference network $\mathcal{C}$ on $\langle L, \preceq, \otimes \rangle$ the set of optimal scores of $\mathcal{C}$ is the Kernel of $Scores(\mathcal{C})$ with respect to $\preceq$, i.e., $a \in Kernel_{\preceq}(Scores(\mathcal{C}))$ iff a is an optimal score for $\mathcal{C}$.*

When $\preceq$ is a total order, then $Kernel_{\preceq}(Scores(\mathcal{C}))$ is the singleton set containing the unique optimal score for $\mathcal{C}$. However, when $\preceq$ is partial, there may be several optimal scores that are pairwise incomparable. Given a preference network $\mathcal{C}$ defined on $\langle L, \preceq, \otimes \rangle$, the size of $Kernel_{\preceq}(Scores(\mathcal{C}))$ is in the worst case equal to the width of $\preceq$, noted $w(\preceq)$. The width of $\preceq$ is the cardinality of the largest subset S of L which only contains incomparable elements (so that $a \not\prec b$ for all $a, b \in S$).

Let $\mathcal{L} = \langle L, \preceq, \otimes \rangle$ be a preference degree structure. We consider the following two problems:

[OPT$_{\mathcal{L}}$]: *Given a preference network $\mathcal{C}$ built on preference degree structure $\mathcal{L}$ and $a \in L$, does there exist an assignment $d \in D_X$ such that $score_{\mathcal{C}}(d) \prec a$.*

[FULLOPT$_{\mathcal{L}}$]: *Given a network $\mathcal{C}$ built on preference degree structure $\mathcal{L}$, and given $H \subseteq L$, does there exist an assignment $d \in D_X$ such that $\exists a \in H, score_{\mathcal{C}}(d) \prec a$.*

Under weak assumptions these problems are NP-complete, in particular, under the conditions specified in the following result.

Proposition 2 *Let $\mathcal{L} = \langle L, \preceq, \otimes \rangle$ be a preference degree structure. Suppose that testing $a \preceq b$ is polynomial, that computing the combination of a multiset of elements of L is polynomial, and that L contains some element a such that $a \succ \mathbf{1}$. Then $OPT_{\mathcal{L}}$ and $FULLOPT_{\mathcal{L}}$ are NP-complete.*

So, the optimization problem in its simple version (find an element of the Kernel) or its full version is not harder in the case of a partially ordered scale than in the case of a totally ordered one.

3.3 Examples of preference networks

Let us present a large class of examples that can be captured by the framework:

• *MAX CSP and VCSP.* In the MAX CSP [10] (resp. VCSPs [21]) framework, the aim is to find an assignment d to all variables that minimizes the number of violated constraints (resp. a combination, for example, the sum, of the weight of the violated constraints). We shall use $L = \mathbb{N} \cup \{+\infty\}$. $\otimes$ is the addition of numbers and $\preceq = \leq$. In these examples, L is totally ordered, $\otimes$ admits a neutral element (i.e., 0) which is the best score is L.

• *Semiring structures,* as those used in semiring CSPs [3], are particular cases of preference degree structures in the sense that they assume the existence of an operator $\oplus$ such that $a \preceq b \iff a \oplus b = a$. If such an operator exists for a preference degree structure, then $\oplus$ is idempotent. Semiring CSPs moreover assume that the unit element of $\otimes$ is absorbing for $\oplus$.

• *Bi-attribute Pareto decision making.* In many multicriteria problems one has to simultaneously optimize several non-commensurable quantities, such as cost, time, security, etc. For instance, in the problem of bi-scaled shortest path [13], each edge in a graph is labeled by a cost and a duration. The cost (resp. the duration) of a path is the sum of the costs (resp. durations) of its edges. For these problems, we can use $L = (\mathbb{N} \cup \{+\infty\}) \times (\mathbb{N} \cup \{+\infty\})$, $\otimes$ being pointwise addition $(a, b) \otimes (a', b') = (a + a', b + b')$. Pairs are compared according to the Pareto rule: $(a, b) \preceq (a', b')$ iff $a \leq a'$ and $b \leq b'$. $\preceq$ is a partial order, but not a total order, e.g., $(3, 2)$ and $(2, 3)$ are incomparable.

• *Order Of Magnitude (OOM) Reasoning.* We describe a slight variation of the system of order of magnitude reasoning defined in [24]. The elements of L are pairs $\langle s, r\rangle$ where $s \in \{+, -, \pm\}$, and $r \in \mathbb{Z} \cup \{-\infty\}$. The system is interpreted in terms of "order of magnitude" values of cost. For example, $\langle -, r\rangle$ represents something which is negative and has order of magnitude K^r (for a large number K). Element $\langle \pm, r\rangle$ arises from the sum of $\langle +, r\rangle$ and $\langle -, r\rangle$. $\langle \pm, r\rangle$ can be thought of as the interval between $\langle -, r\rangle$ and $\langle +, r\rangle$, since the sum of a positive quantity of order K^r and a negative quantity of order K^r can be either positive or negative and of any order less than or equal to r. For these problems, $L = \{\langle \pm, -\infty\rangle\} \cup \{\langle s, r\rangle \,:\, s \in \{+, -, \pm\}, r \in \mathbb{Z}\}$, and the interpretation leads us to define the summation operation $\otimes$ by:

$$\langle s, r\rangle \otimes \langle s', r'\rangle = \begin{cases} \langle s, r\rangle, & if \;\; r > r'; \\ \langle s', r'\rangle, & if \;\; r < r'; \\ \langle s \vee s', r\rangle, & if \;\; r = r'; \end{cases}$$

where $\vee$ is given by: $+ \vee + = +$ and $- \vee - = -$, and otherwise, $s \vee s' = \pm$. Operation $\otimes$ is commutative and associative with neutral element $\langle \pm, -\infty\rangle$. $\preceq$ is defined by the following instances: (i) for all r and s, $\langle -, r\rangle \preceq \langle +, s\rangle$; (ii) for all $s \in \{+, -, \pm\}$, and all r, r' with $r \geq r'$: $\langle -, r\rangle \preceq \langle s, r'\rangle \preceq \langle +, r\rangle$. $\preceq$ is a partial order, but not a total order; for example, $\langle \pm, r\rangle$ and $\langle \pm, s\rangle$ are incomparable when $r \neq s$.

• *Tolerant Pareto.* The problem with a Pareto-based comparison is that the preference provided is often not decisive enough. For instance, the two pairs $a = (a_{cost}, a_{time})$ and $b = (b_{cost}, b_{time})$ are incomparable as soon as $a_{cost} < b_{cost}$ and $b_{time} < a_{time}$, and this even if the difference between a_{cost} and b_{cost} is much greater than difference between b_{time} and a_{time}.

Consider our time/cost pair. The idea is to use indifference thresholds, say α_{cost} for the first dimension, and α_{time} for the second one. If $a_{cost} + \alpha_{cost} < b_{cost}$, we shall say that the cost dimension has a strong preference for a over b, and opposes a veto to the opposite preference. Then we decide that an alternative is better than the other iff it Pareto dominates, but with respect to the thresholds of tolerance. Formally,

$$\begin{aligned} a \prec b \iff & (b_{cost} - a_{cost} > \alpha_{cost} \wedge b_{time} - a_{time} \geq -\alpha_{time}) \\ & \vee (b_{time} - a_{time} > \alpha_{time} \wedge b_{cost} - a_{cost} \geq -\alpha_{cost}) \\ a \sim b \iff & a = b \end{aligned}$$

So, when one dimension strongly prefers alternative a while the other does not oppose a veto we do not get an incomparability, like in the classical Pareto case, but a strict preference $a \prec b$. This decision rule is related to the Electre method (see e.g. [20]). It yields a preference relation that is not complete nor transitive: it may happen that $a \prec b$ and $b \prec c$ while a and c are not comparable (e.g. because the time dimension that does not oppose a veto to $a \prec b$ nor to $b \prec c$ is a vetoer for $a \prec c$). Nevertheless, $\prec$ is acyclic.

This example cannot be cast as a preference network *stricto sensu*, but its closure by transitivity can be, using pointwise addition as the combination. Let $\prec^*$ be the transitive closure of $\prec$. It can be shown that $a \prec^* b$ holds if and only if either (i) $b_{cost} - a_{cost} > 0$ and $b_{time} - a_{time} > 0$, or (ii) there exists $k \in \{1, 2, \ldots\}$ such that either (a) $b_{cost} - a_{cost} > k\alpha_{cost}$ and $b_{time} - a_{time} \geq -k\alpha_{time}$ or (b) $b_{time} - a_{time} > k\alpha_{time}$ and $b_{cost} - a_{cost} \geq -k\alpha_{cost}$.

In this rule, the thresholds are considered as elementary units of strong preference. So, a is better than b when, going from b to a, the enhancement on one dimension (e.g. the cost dimension) is greater than the degradation in the other dimension, this enhancement (resp. degradation) being evaluated on a scale whose unit is α_{cost} (resp. α_{time}).

4 Casting preference networks in the local computation scheme

Applications of local computation for optimization focus on the case when the optimization is made with respect to a total order (though see [19, 15]). We will show that it applies to many other situations, which involve only partially ordered scales.

In the following, we present three ways of embedding preference networks into Shenoy and Shafer's framework in order to benefit from the local computation machinery. It is important to note that in all these encodings our focus in on optimization problems using partial orders. First, we show that a direct encoding of the preference degree structure is inadequate, because the local computation algorithm will only generate the greatest lower bound of the achievable scores, rather than an optimal score. Then, we investigate two alternative approaches: the use of an extension of the original order (this provides one of the optimal scores, provided that such an extension exists), and the use of a set encoding of the preference degree structure (this is always possible and provides all the optimal scores).

4.1 Direct encoding

Preference networks can be simply cast as a problem of combination of valuations, letting $\mathcal{V} = \bigcup_{S \subseteq X} \{f : D_S \mapsto L\}$ and defining $\boxtimes$ in a pointwise fashion:

Definition 5 *Let $\langle L, \preceq, \otimes \rangle$ be a preference degree structure, let X be a set of variables, and let σ and τ be two local functions. The local function $\sigma \boxtimes \tau$ with scope $scope(\sigma) \cup scope(\tau)$ is defined as follows. For any $d \in D_{scope(\sigma) \cup scope(\tau)}$, define $(\sigma \boxtimes \tau)(d) = \sigma(d) \otimes \tau(d)$. (Recall that $\sigma(d)$ means $\sigma(d[scope(\sigma)])$.)*

Then the global score function is simply the combination of the c_i in $\mathcal{C}$.

Proposition 3 *For any preference network $\mathcal{C}$ over $\langle L, \preceq, \otimes\rangle$ and $d \in D_X$, $score_{\mathcal{C}}(d) = (\boxtimes_{c_i \in \mathcal{C}} c_i)(d)$.*

Also, $\boxtimes$ satisfies axiom $A2$ iff $\otimes$ is associative and commutative, which gives a fundamental justification for having $\otimes$ associative and commutative in preference degree structures.

Example 1. Consider a MAX CSP problem with two variables $X = \{x_1, x_2\}$ with domains $D_1 = D_2 = \{0, 1\}$. Recall that the preference degree structure in MAX CSP problems is $\langle L = \mathbb{N} \cup \{\infty\}, \preceq = \leq, \otimes = +\rangle$. Let $\mathcal{C} = \{c_1, c_2\}$, where $c_1(x_1) = x_1$ and $c_2(x_2) = x_2$. The global score function $c_1(x_1) \otimes c_2(x_2)$ is a new function with scope $\{x_1, x_2\}$ which can be extensionally defined with the following table:

x_1	x_2	$c_1(x_1) \otimes c_2(x_2)$
0	0	0
0	1	1
1	0	1
1	1	2

Now, consider a bi-attribute Pareto decision making problem defined over the same set of variables and domain values as the previous example. Recall that the preference degree structure in this case is $\langle L = (\mathbb{N} \cup \{+\infty\}) \times (\mathbb{N} \cup \{+\infty\}), \preceq, \otimes\rangle$, where $\otimes$ is pointwise addition and $\preceq$ is the simple Pareto comparison. Let $\mathcal{C} = \{c_1, c_2\}$, where $c_1(x_1) = (x_1, 1 - x_1)$ and $c_2(x_2) = (x_2, x_2)$. The global score function $c_1(x_1) \otimes c_2(x_2)$ is a new function with scope $\{x_1, x_2\}$ which can be extensionally defined with the following table:

x_1	x_2	$c_1(x_1) \otimes c_2(x_2)$
0	0	(0,1)
0	1	(1,2)
1	0	(1,0)
1	1	(2,1)

■

We will see below that difficulties arise with the marginalisation operator when $\preceq$ is not a total order. For the case when $\preceq$ is a total order, the min operator is well defined and we can set, for $T \subseteq scope(\sigma)$,

$$\sigma^{\downarrow T}(d) = min_{d' \in D_{scope(\sigma)}, d = d'[T]}\{\sigma(d')\}$$

This definition ensures the satisfaction of axioms A1 and A3, and that for any $\mathcal{C}$ built on $\langle L, \preceq, \otimes\rangle$, $(\boxtimes_{c \in \mathcal{C}} c)^{\downarrow\emptyset}$ is the optimal score for $\mathcal{C}$.

We can consider using the same technique when there exists an operator $\oplus$ such that $a \preceq b \iff a \oplus b = a$, and such that $\langle L, \otimes, \oplus\rangle$ is a (commutative) semiring, i.e., $\otimes$ and $\oplus$ are both associative and commutative, and $\otimes$ distributes over $\oplus$: $a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$ for all $a, b, c \in L$. This assumption is made in semiring CSPs [2, 3]. We can then define a marginalization operator as follows:

Definition 6 *Suppose there exists a commutative and associative operator $\oplus$ such that $a \preceq b \iff a \oplus b = a$, and $\otimes$ distributes over $\oplus$. Let us define the operation $\downarrow$ as follows, where T is any subset of scope(σ):*

$$\forall \sigma, d \in D_T : \sigma^{\downarrow T}(d) = \bigoplus_{d' \in D_{scope(\sigma)}, d = d'[T]} \sigma(d')$$

The local computation axioms are then satisfied (the result follows from Theorem 2 of [15]):

Proposition 4 *Let $\langle L, \preceq, \otimes \rangle$ be a preference degree structure. Suppose that there exists a commutative and associative operator $\oplus$ such that $a \preceq b \iff a \oplus b = a$, and such that $\otimes$ distributes over $\oplus$. Then, axioms A1, A2 and A3 are satisfied by $\boxtimes$ and $\downarrow$ as defined in Definitions 5 and 6.*

It follows that under the conditions of Proposition 4 we can use sequential elimination to compute $(\boxtimes_{c \in \mathcal{C}} c)^{\downarrow \emptyset}$. Unfortunately, this computation is not faithful to the notion of optimality in L: it may well happen that the score computed by this marginalisation is not achievable: $(\boxtimes_{c \in \mathcal{C}} c)^{\downarrow \emptyset}$ *does not necessarily belong to the kernel at all.* More precisely, the following holds:

Theorem 1 *Given a preference degree structure $\langle L, \preceq, \otimes \rangle$, and if $\boxtimes$ and $\downarrow$ are defined according to Definitions 5 and 6, the following assertions are equivalent:*

- $\forall \mathcal{C}, (\boxtimes_{c \in \mathcal{C}} c)^{\downarrow \emptyset} \in Kernel_{\preceq}(Scores(\mathcal{C}))$
- $\preceq$ *is a total order.*

This result is a rather negative one: unless we are working with a total order, the computation of a (sub)set of optimal scores cannot be understood in terms of combination and marginalization of soft constraints as it is usually done. What this kind of approach computes is actually a (greatest) lower bound of the Kernel:

Proposition 5 *If $\boxtimes$ and $\downarrow$ are defined according to Definitions 5 and 6, then $\forall a \in Kernel_{\preceq}(Scores(\mathcal{C}))$, $(\boxtimes_{c_i \in \mathcal{C}} c_i)^{\downarrow \emptyset} \preceq a$. In fact, $(\boxtimes_{c_i \in \mathcal{C}} c_i)^{\downarrow \emptyset}$ is the greatest lower bound of $Kernel_{\preceq}(Scores(\mathcal{C}))$.*

However, this value $(\boxtimes_{c_i \in \mathcal{C}} c_i)^{\downarrow \emptyset}$ does not necessarily belong to the kernel.

Example 2. Consider the problems defined in Example 1. The operator $\oplus$ in the MAX CSP problem is min. It is easy to see that the kernel of the set of scores is 0, which can be obtained by marginalizing the global score function over all variables. The operator $\oplus$ in the bi-attribute Pareto decision making problem is the pointwise minimum, i.e., $(a, b) \oplus (a', b') = (min(a, a'), min(b, b'))$. The kernel of the set of scores is $\{(0,1),(1,0)\}$. However, $(c_1 \otimes c_2)^{\downarrow\emptyset} = (0,0)$, although there is no assignment in D_X with a global score of $(0,0)$. Note that $(0,0)$ is a lower bound of $\{(0,1),(1,0)\}$, that is, $(0,0) \preceq (0,1)$ and $(0,0) \preceq (1,0)$. ■

Finally, recall that variable elimination approaches are potentially exponential in time and space with respect to the treewidth, which is rather computationally expensive for just an approximation of the result. We shall circumvent this difficulty by working with another comparator. The first solution is to simply extend $\preceq$.

4.2 Extending $\preceq$

A classical approach in Pareto-based multicriteria optimization problems is to optimize a linear combination of the criteria. The important idea here is that any solution minimizing this sum is known to be Pareto optimal. Namely, one optimizes according to a new comparator, say $\preceq'$, such that $a \preceq b$ implies $a \preceq' b$: if a is better than b according to the original relation, then it is still the case with the new one. But $\preceq'$ can rank scores that are incomparable with respect to $\preceq$. Such a relation is called an extension of the original relation.

Definition 7 *Let $\preceq'$ and $\preceq$ be two relations on a set L. Then we say $\preceq'$* extends *$\preceq$ if and only if $a \preceq b$ implies $a \preceq' b$.*

Example 3. Consider our running bi-attribute Pareto decision making problem in Example 1. Let us interpret valuation (a, b) as $(cost, time)$. Then, we shall decide $(a, b) \preceq' (a', b')$ if and only if $a + \beta \cdot b \leq a' + \beta \cdot b'$. $\preceq'$ is complete and if β is high enough, there are no ties, so that $\preceq'$ is a total order (equal to a lexicographic order with time being more important than cost). ■

Optimizing with respect to an extension leads to solutions that are optimal with respect to the original relation. More precisely:

Proposition 6 *Let $\preceq'$ and $\preceq$ be two partial orders on a set L. If $\preceq'$ extends $\preceq$, then for any $A \subseteq L$, $Kernel_{\preceq'}(A) \subseteq Kernel_{\preceq}(A)$.*

Now, if there exists a totally ordered extension $\preceq'$ of $\preceq$ such that $\langle L, \preceq', \otimes \rangle$ is a preference degree structure, it is then possible to define $\oplus$ as the min of two scores according to $\preceq'$.

Like the approach described in Section 4.1, the present one provides the user with a unique score, but this one has the advantage of systematically proving an optimal one, rather than a lower bound. As a matter of fact, we have seen that when optimizing with respect to Pareto, we can use an extension based on a weighted sum (or a lexicographic order). The OOM comparison can also be extended by such a comparison procedure, using super increasing weights [6].

Unfortunately, such a totally ordered extension does not necessarily exist, as shown in the following example.

Example 4. Consider a preference network defined over the preference degree structure $\langle L = \mathbb{Z}, \preceq, \otimes = \times \rangle$, where $a \preceq b \iff a = b$. Suppose there exists a total order $\preceq'$ on L extending $\preceq$ and such that $\langle L, \preceq', \otimes \rangle$ is a preference degree structure. Since $\preceq'$ is total, we have either $1 \preceq' -1$ or $-1 \preceq' 1$. $1 \preceq' -1$ implies by monotonicity $1 \otimes -1 \preceq' -1 \otimes -1$, and so $-1 \preceq' 1$. Similarly, $-1 \preceq' 1$ implies $1 \preceq' -1$, so in either case we have $1 \preceq' -1 \preceq' 1$, contradicting antisymmetry. ■

The following result gives sufficient conditions for an appropriate extension to exist, where a^1 is defined to be a and, for $k \geq 1$, $a^{k+1} = a^k \otimes a$.

Theorem 2 *Let $\langle L, \preceq, \otimes \rangle$ be a preference degree structure with unit element $\mathbf{1}$, which also satisfies the following two properties:*

(i) for all $a, b \in L$ with $a \neq b$ and all $k > 0$ we have $a^k \neq b^k$;
(ii) $a \otimes c \preceq b \otimes c \Rightarrow a \preceq b$ for all $a, b, c \in L$.

Then there exists a total order $\preceq'$ on L extending $\preceq$ and such that for all $a, b, c \in L$, $a \otimes c \preceq' b \otimes c \iff a \preceq' b$, and so, in particular, $\langle L, \preceq', \otimes \rangle$ is a preference degree structure.

The idea behind the proof of the theorem is, roughly speaking, that, unless $\preceq$ is already a total order, we can always add extra orderings to $\preceq$ whilst maintaining the properties (i) and (ii) and the properties of a preference degree structure. A maximal relation satisfying the properties and extending $\preceq$ is then a total order.

The theorem requires strong conditions. Condition (ii), together with the monotonicity property of preference degree structures, implies the following strong monotonicity property: $a \otimes c \preceq b \otimes c \iff a \preceq b$ for all $a, b, c \in L$. This implies, in particular, that if $a \neq b$ then for all c, $a \otimes c \neq b \otimes c$. Condition (i) is a somewhat similar property to this. For an example where the conditions are satisfied, consider the preference degree structure $\langle L, \preceq, \otimes \rangle$, similar to that described for bi-attribute Pareto decision making, with $L = \mathbb{N} \times \mathbb{N}$, $\otimes$ being pointwise addition $(a, b) \otimes (a', b') = (a + a', b + b')$, and where $\preceq$ is given by: $(a, b) \preceq (a', b') \iff a \leq a'$ and $b \leq b'$.

Because $\langle L, \preceq', \otimes \rangle$ is a preference degree structure based on a total order, it is possible to define $\oplus$ as the min of two scores according to $\preceq'$. Then, Definitions 5 and 6 can be applied and Axioms A1, A2 and A3 are satisfied, thanks to Proposition 4. Hence the sequential elimination procedure (see Section 2) may be used to obtain an optimal score with respect to $\preceq'$, which is also (because $\preceq'$ extends $\preceq$) an optimal score with respect to $\preceq$. In other words, when conditions (i) and (ii) are satisfied, it is always possible to obtain an optimal score by non-serial dynamic programming.

4.3 Set-based Encoding

There is a very general way of using Shafer-Shenoy framework to optimize over a preference degree structure. The idea is to move from L to 2^L (i.e., the set of subsets of L) and define a marginalization operator able to keep the best scores in L according to $\preceq$. In other words, when a and b in L are not comparable, the marginalization operator keeps both. Then, if a preference network $\mathcal{C}$ defined over L is mapped to one over 2^L, local computation over the new network is faithful to the notion of optimality in L.

There exist other works in the literature aiming to use local computations over optimization problems using partial orders. In [19, 8], set-based encoding is defined as a way to deal with multiobjective optimization problems based on a Pareto comparison. The transformation has been defined over c-semirings by Rollon and Larrosa [18] under the name of *frontier algebra*. Our encoding generalises the latter. In short, we show that (i) the set-based encoding also works in the general case of any type of partially ordered scoring scale and (ii) preference networks are rich enough to capture this kind of transformation.

We define the set-based encoding extension of a preference degree structure as follows.

Definition 8 *Let $\langle L, \preceq, \otimes \rangle$ be a preference degree structure. Then, its set-based encoding extension is $\langle \mathbf{L}, \otimes_s, \oplus_s \rangle$, where*

- $\mathbf{L} = \{\text{finite } A \subseteq L : A \neq \emptyset, A = Kernel_{\preceq}(A)\}$
- $A \otimes_s B = Kernel_{\preceq}(\{a \otimes b : a \in A, b \in B\})$
- $A \oplus_s B = Kernel_{\preceq}(A \cup B)$.

In words, $\mathbf{L}$ is the set of finite subsets of L that do not contain comparable elements according to $\preceq$, and $\otimes_s$ and $\oplus_s$ are able to keep all non-comparable elements according to $\preceq$ in $\mathbf{L}$. Notice that a singleton is its own kernel, thus belongs to $\mathbf{L}$. Moreover, $\mathbf{L}$ is stable with respect to the kernel-based union: for any $A, B \in \mathbf{L}$, $Kernel_{\preceq}(A \cup B) \in \mathbf{L}$.

We have the following important property:

Proposition 7 *$\langle \mathbf{L}, \otimes_s, \oplus_s \rangle$ is a (commutative) semiring.*

The $\oplus_s$ operator is faithful to $\preceq$, in the following sense:

Proposition 8 *$\forall a, b \in L,\ a \preceq b \iff \{a\} \oplus_s \{b\} = \{a\}$*

Let $\mathcal{C}$ be a preference network defined over the preference degree structure $\langle L, \preceq, \otimes \rangle$. For any local function c in $\mathcal{C}$, let $\mathbf{c}$ be the local function taking its scores in $\mathbf{L}$, defined by: $\mathbf{c}(d) = \{c(d)\}$ and denote $\mathbf{C} = \{\mathbf{c} : c \in \mathcal{C}\}$ the transformation of $\mathcal{C}$ by this "singletonization". We then have:

Proposition 9 *Axioms A1, A2 and A3 are satisfied by $\boxtimes$ and $\downarrow$ as defined in Definitions 5 and 6 from the set operations $\otimes_s$ and $\oplus_s$ provided by Definition 8.*

Thanks to Proposition 9, we can then compute $(\boxtimes_{\mathbf{c} \in \mathbf{C}} \mathbf{c})^{\downarrow \emptyset}$ using local computation. $(\boxtimes_{\mathbf{c} \in \mathbf{C}} c)^{\downarrow \emptyset}$ provides a unique score in the set based preference degree scale $\mathbf{L}$. We shall then prove that this set is the set of scores in L that are optimal with respect to $\preceq$.

Proposition 10 $(\boxtimes_{\mathbf{c}\in\mathbf{C}}\ c)^{\downarrow\emptyset} = Kernel_{\preceq}(Scores(\mathcal{C}))$.

In other words, the set of optimal elements $Kernel_{\preceq}(Scores(\mathcal{C}))$ can be expressed as the marginalization of a combination, and computed using local computation. Through the set-based encoding, local computation can be used to compute the set of optimal values of any preference network, i.e., variable elimination is possible for *any* preference network.

Example 5. Consider the bi-attribute Pareto decision making problem in Example 1. Recall that the set of variables is $X = \{x_1, x_2\}$, the set of domain values is $D_1 = D_2 = \{0, 1\}$, and the set of functions is $\mathcal{C} = \{c_1, c_2\}$ where $c_1(x_1) = (x_1, 1 - x_1)$ and $c_2(x_2) = (x_2, x_2)$. The singletonization of $\mathcal{C}$ is $\mathbf{C} = \{\mathbf{c_1}, \mathbf{c_2}\}$, where $\mathbf{c_1}(x_1) = \{(x_1, 1 - x_1)\}$ and $\mathbf{c_2}(x_2) = \{(x_2, x_2)\}$. The global score function $\mathbf{c_1}(x_1) \otimes_s \mathbf{c_2}(x_2)$ is a new function with scope $\{x_1, x_2\}$ which can be extensionally defined with the following table:

x_1	x_2	$\mathbf{c_1}(x_1) \otimes_s \mathbf{c_2}(x_2)$
0	0	{(0,1)}
0	1	{(1,2)}
1	0	{(1,0)}
1	1	{(2,1)}

The result of computing $(\mathbf{c_1} \otimes_s \mathbf{c_2})^{\downarrow\emptyset}$ is $\{(0, 1), (1, 0)\}$, which is the kernel of $Scores(\mathcal{C})$, i.e., the set of optimal scores. ■

The theoretical application of local computation must not overshadow its practical range of application. As we have said, variable elimination is in the worst case exponential with respect to the structural parameter treewidth. This is the case if we consider that size of the score sets is 1. Depending on how discriminating $\preceq$ is, we may get a considerably larger score set at some point in the computation.

We assume that any operation $a \otimes b$ or comparison $a \preceq b$ is made in bounded constant time.

Proposition 11 *Given a preference network $\mathcal{C}$ on any preference degree structure $\langle L, \preceq, \otimes\rangle$, the time and space complexity of the variable elimination procedure is $O(w(\preceq)^2 \times (m + n) \times d^{tw})$ and $O(w(\preceq) \times n \times d^{tw})$, respectively, where $w(\preceq)$ is the width of the order $\preceq$, m is the number of functions, n is the number of variables, d is the maximum domain size and tw is the structural parameter treewidth.*

The width of the order $\preceq$ can, in certain cases be relatively small. Its value is 1, obviously, for the total orders (e.g. Max CSP). For the OOM (Order of Magnitude reasoning) case, the largest kernel is $\{\langle\pm, r_1\rangle, \ldots, \langle\pm, r_k\rangle\}$, where $\{r_1, \ldots, r_k\}$ is the set of possible values for the order of magnitude (typically, reduced to a small selection of qualitative values: “null”, negligible”, “weak”, "significant", "high", "very high"). For Pareto comparison on n criteria, all of them using a totally ordered scale the width is exponential in the number of criteria. This is an additional reason to prefer extensions of the partial order where possible, for example using a weighted sum as usually done, or to use partial orders as extensions; one might also consider, for example, a tolerant Pareto rule or attempting to use an epsilon approximation [8, 17] to reduce the number of incomparabilities (although the indifference relation is no longer transitive).

However, it is important to note that there exist other important practical cases as, for example, Pareto comparison based on a product of totally ordered scales such as integers or reals, for which the width of the order is not finite. In such cases, the sizes of the kernels can become very large, and local computation can be impractical.

5 Conclusion and perspectives

The Shafer-Shenoy framework characterizes the assumptions under which local computation can be applied. Although the framework does not restrict the order among scores to be total, the main applications of local computation are based on total orders. However, the practical importance of problems using partial orders is undeniable, for example, in multi-criteria optimization.

In this paper we propose an algebraic structure called preference network able to capture a great variety of problems, and focus on the ways of embedding preference networks into the Shafer-Shenoy framework. We have shown that a direct encoding is sound with respect to optimality when using total orders, but it is not necessarily the case when using partial orders. To overcome this difficulty, we propose two alternative encodings. The first one is based on extending the partial order into a total one. Its shortcoming however is that in some problems a total order extension does not necessarily exist. Thus, this approach is not always applicable. The second approach is based on a set-based encoding of the original network. The virtue of this approach is that it is always possible. Therefore, any preference network, using either a total or a partial order, can always benefit from the local computation machinery through our set-based encoding, although there can be computational problems, in particular, if the number of optimal solutions is large.

But, as is the case for the tolerant Pareto example, there are meaningful structures of preferences that are not captured by preference networks. Other examples include preorders and semiorders, that allow richer indifference relations than identity, and can lead to less incomparabilities than if one attempted to model the problem with a partial order. Further research will be developed around the algebraic study of such structures.

Appendix: Proofs

This appendix includes proofs of the results, with the second part consisting of the proof of Theorem 2 along with some additional auxiliary results.

Proof (Proposition 1).

Suppose that $a \in Kernel_{\preceq}(Scores(\mathcal{C}))$. Then there exists some $d \in D_X$ with $a = score_{\mathcal{C}}(d)$. If a were not an optimal score for $\mathcal{C}$ then d would not be an optimal solution so there would exist d' with $score_{\mathcal{C}}(d') \prec score_{\mathcal{C}}(d) = a$, which would contradict $a \in Kernel_{\preceq}(Scores(\mathcal{C}))$ because $score_{\mathcal{C}}(d') \in Scores(\mathcal{C})$. Hence $a \in Kernel_{\preceq}(Scores(\mathcal{C}))$ implies a is an optimal score for $\mathcal{C}$.

Conversely, suppose that a is an optimal score for $\mathcal{C}$. Then $a = score_{\mathcal{C}}(d)$ for some optimal solution d, so $a \in Scores(\mathcal{C})$. If $a \notin Kernel_{\preceq}(Scores(\mathcal{C}))$ then there exists some $a' \in Scores(\mathcal{C})$ with $a' \prec a$, and so there exists some $d' \in D_X$ with $a' = score_{\mathcal{C}}(d')$, which contradicts the fact that d is an optimal solution. Hence if a is an optimal score for $\mathcal{C}$ then $a \in Kernel_{\preceq}(Scores(\mathcal{C}))$.

■

Proof (Proposition 2).

The fact that $\mathrm{OPT}_{\mathcal{L}}$ and $\mathrm{FULLOPT}_{\mathcal{L}}$ are in NP follows easily, since, by the hypothesis, combination and testing of dominance are polynomial, so we can confirm in polynomial time that $score_{\mathcal{C}}(d) \prec a$ for any given d and a.

To prove NP-hardness of $\mathrm{OPT}_{\mathcal{L}}$: By the hypothesis, there exists some element a such that $a \succ \mathbf{1}$. It follows by monotonicity that $a^k \succeq a^{k-1}$ for all $k > 0$, where $a^1 = a$ and $a^k = a \otimes a^{k-1}$, for $k > 1$. Thus, by transitivity, $a^k \succeq a$ for all $k > 0$ and thus, using transitivity again, $a^k \succ \mathbf{1}$ for all $k > 0$.

We can use a reduction from 3SAT, by generating for each clause a local function on its variables which only take two different values: $\mathbf{1}$ (when the clause is satisfied) and a (when violated). Then for any d, either $score(d) = \mathbf{1}$ (when d satisfies all the clauses) or $score(d) \succeq a$ (when d violates some clause). Hence, the CNF is satisfiable if and only if there is a d such that $score(d) \prec a$. This proof also shows that $\mathrm{FULLOPT}_{\mathcal{L}}$ is NP-hard (letting $H = \{a\}$).

■

Proof (Proposition 3). This follows immediately from Definitions 3 and 5. ■

Proof (Proposition 4).

Axiom $A1$ can be shown to hold, using the commutativity and associativity of $\oplus$. Axiom $A2$ follows immediately from the commutativity and associativity of $\otimes$. Axiom $A3$ can be proved using distributivity: see the proof of Theorem 2 in [15], noticing that $\langle L, \otimes, \oplus \rangle$ is a semiring. ■

Proof (Theorem 1).

By the definitions, $(\boxtimes_{c \in \mathcal{C}} c)^{\downarrow \emptyset}$ is equal to $\bigoplus(Scores(\mathcal{C}))$. Suppose firstly that $\preceq$ is a total order. Then $\oplus$ is just minimum with respect to $\preceq$, so $\bigoplus(Scores(\mathcal{C}))$ is the minimum element of $Scores(\mathcal{C})$. Thus $(\boxtimes_{c \in \mathcal{C}} c)^{\downarrow \emptyset}$ is the unique element in $Kernel_{\preceq}(Scores(\mathcal{C}))$.

Conversely, suppose that $\preceq$ is not a total order, so that there are two scores a and b that are incomparable, i.e., $a \oplus b \neq a, b$. Then for the trivial problem $\mathcal{C} = \{c\}$ with $c(1) = a$, $c(2) = b$, $\mathit{Kernel}_{\preceq}(\mathit{Scores}(\{c\})) = \{a, b\}$, whereas $c^{\downarrow\emptyset}$ equals $a \oplus b$, which is different from a and from b. ■

Proof (Proposition 5).
We first show that (i) for any finite $A \subseteq L$, the greatest lower bound of A is equal to $\bigoplus A$. The equivalence $a \preceq b \iff a \oplus b = a$ and idempotence of $\oplus$ easily imply that $\bigoplus A$ is a lower bound for A. Now, consider any lower bound b for A. We have that $b \preceq a_1$ and $b \preceq a_2$ implies that $b \preceq a_1 \oplus a_2$, using the above equivalence. Iterating this yields $b \preceq \bigoplus A$. This implies that $\bigoplus A$ is the greatest lower bound for A.

Next we show that (ii) $\bigoplus A$ is equal to $\bigoplus \mathit{Kernel}_{\preceq}(A)$. For any $a \in A \setminus \mathit{Kernel}_{\preceq}(A)$, there exists $b \in \mathit{Kernel}_{\preceq}(A)$ with $b \preceq a$. Then $a \oplus b = b$, which implies that $\bigoplus A$ equals $\bigoplus A \setminus \{a\}$. Iterating this shows that $\bigoplus A$ is equal to $\bigoplus \mathit{Kernel}_{\preceq}(A)$.

Together, (i) and (ii) imply that the greatest lower bound of $\mathit{Kernel}_{\preceq}(A)$ is $\bigoplus \mathit{Kernel}_{\preceq}(A)$ which equals $\bigoplus A$. Putting $A = \mathit{Scores}(\mathcal{C})$, we have that $(\boxtimes_{c_i \in \mathcal{C}} c_i)^{\downarrow\emptyset}$, which equals $\bigoplus A$, is equal to the greatest lower bound of $\mathit{Kernel}_{\preceq}(\mathit{Scores}(\mathcal{C}))$. ■

Proof (Proposition 6). This follows immediately from Definition 7. ■

To prove Proposition 7 we make use of the following lemma:

Lemma 1. *Let A, B and C be finite subsets of L. We write $A \otimes' B$ for $\{a \otimes b : a \in A, b \in B\}$. We abbreviate $\mathit{Kernel}_{\preceq}$ to Kernel.*

(i) Operation $\otimes'$ is commutative and associative;
(ii) $A \otimes' (B \cup C) = (A \otimes' B) \cup (A \otimes' C)$.
(iii) $\mathit{Kernel}(A \cup (\mathit{Kernel}(B)) = \mathit{Kernel}(A \cup B)$;
(iv) $\mathit{Kernel}(A \otimes' (\mathit{Kernel}(B)) = \mathit{Kernel}(A \otimes' B)$;

Proof. (i) Commutativity of $\otimes'$ follows from commutativity of $\otimes$. Regarding associativity, we have $A \otimes' (B \otimes' C) = A \otimes' \{b \otimes c : b \in B, c \in C\} = \{a \otimes (b \otimes c) : a \in A, b \in B, c \in C\}$. Similarly, we can see that $(A \otimes' B) \otimes' C = \{(a \otimes b) \otimes c : a \in A, b \in B, c \in C\}$. Associativity of $\otimes'$ then follows from that of $\otimes$,

(ii) $A \otimes' (B \cup C) = \{a \otimes d : d \in B \cup C\} = \{a \otimes b : b \in B\} \cup \{a \otimes c : c \in C\} = (A \otimes' B) \cup (A \otimes' C)$.

(iii) We first show that $\mathit{Kernel}(A \cup (\mathit{Kernel}(B))) \subseteq \mathit{Kernel}(A \cup B)$. To prove a contradiction, suppose that $a \in \mathit{Kernel}(A \cup (\mathit{Kernel}(B)))$ but $a \notin \mathit{Kernel}(A \cup B)$. We have $a \in A \cup B$, so there exists $b \in A \cup B$ with $b \prec a$. It cannot be the case that $b \in A \cup \mathit{Kernel}(B)$ or else a would not be in $\mathit{Kernel}(A \cup (\mathit{Kernel}(B)))$. Thus $b \in B - \mathit{Kernel}(B)$ so there exists some $c \in \mathit{Kernel}(B)$ with $c \prec b$. Transitivity of $\prec$ implies that $c \prec a$, contradicting $a \in Kernel(A \cup (Kernel(B)))$, since $c \in A \cup (Kernel(B))$.

We next show that $Kernel(A \cup B) \subseteq Kernel(A \cup (Kernel(B)))$. Suppose that $a \in Kernel(A \cup B)$, and to prove a contradiction, that $a \notin Kernel(A \cup (Kernel(B)))$. Since $A \cup Kernel(B) \subseteq A \cup B$ this must mean that $a \notin A \cup Kernel(B)$ (else a wouldn't be in $Kernel(A \cup B)$), so $a \in B - Kernel(B)$. But then there exists $b \in B$ (and so $b \in A \cup B$) with $b \prec a$, which contradicts $a \in Kernel(A \cup B)$.

(iv) We first show that $Kernel(A \otimes' (Kernel(B))) \subseteq Kernel(A \otimes' B)$. Suppose, to prove a contradiction, that $e \in Kernel(A \otimes' (Kernel(B)))$ but $e \notin Kernel(A \otimes' B)$. There exists $a \in A$ and $b \in Kernel(B)$ with $e = a \otimes b$. Since $e \notin Kernel(A \otimes' B)$ there exists $a' \in A$ and $b' \in B$ with $a' \otimes b' \prec e$. Since $b' \in B$ there exists $b'' \in Kernel(B)$ with $b'' \preceq b'$. By monotonicity, $a' \otimes b'' \preceq a' \otimes b' \prec e$, which contradicts $e \in Kernel(A \otimes' (Kernel(B)))$.

We now show that $Kernel(A \otimes' B) \subseteq Kernel(A \otimes' (Kernel(B)))$. Suppose, to prove a contradiction, that there exists some $e \in Kernel(A \otimes' B) - Kernel(A \otimes' (Kernel(B)))$. Write $e = a \otimes b$, where $a \in A$ and $b \in B$. There exists $b' \in Kernel(B)$ with $b' \preceq b$. Monotonicity implies that $a \otimes b' \preceq e$, which implies that $a \otimes b' = e$, since $e \in Kernel(A \otimes' B)$. Hence $e \in A \otimes' Kernel(B)$. Since $e \notin Kernel(A \otimes' (Kernel(B)))$ there exists $f \in Kernel(A \otimes' (Kernel(B)))$ with $f \prec e$. But $f \in A \otimes' B$, which contradicts $e \in Kernel(A \otimes' B)$. ∎

Proof (Proposition 7).

Consider any finite subsets $A, B, C \subseteq L$.

Commutativity of $\oplus_s$: It follows from commutativity of $\otimes'$ (Lemma 1(i)).

Associativity of $\oplus_s$: $A \oplus_s (B \oplus_s C) = Kernel(A \cup Kernel(B \cup C))$, which, by Lemma 1(iii), equals $Kernel(A \cup B \cup C)$. By commutativity of $\oplus_s$, we have $(A \oplus_s B) \oplus_s C = C \oplus_s (A \oplus B)$ which thus equals $Kernel(A \cup B \cup C)$, proving associativity of $\oplus_s$.

Associativity of $\otimes_s$: $A \otimes_s (B \otimes_s C) = Kernel(A \otimes' Kernel(B \otimes' C))$, which by Lemma 1(iv) equals $Kernel(A \otimes' (B \otimes' C))$. Similarly, $(A \otimes_s B) \otimes_s C = Kernel((A \otimes' B) \otimes' Kernel(C)) = Kernel((A \otimes' B) \otimes' C)$. Associativity of $\otimes_s$ then follows from associativity of $\otimes'$ (Lemma 1(i)).

Distributivity: $A \otimes_s (B \oplus_s C) = Kernel(A \otimes' Kernel(B \cup C))$, which equals $Kernel(A \otimes' (B \cup C))$ by Lemma 1(iv), which equals $Kernel((A \otimes' B) \cup (A \otimes' C))$, by Lemma 1(ii).

We have $(A \otimes_s B) \oplus_s (A \otimes_s C) = Kernel(Kernel(A \otimes' B) \cup Kernel(A \otimes' C))$ which equals, applying Lemma 1(iii) twice, $Kernel((A \otimes' B) \cup (A \otimes' C))$. We therefore have $A \otimes_s (B \oplus_s C) = (A \otimes_s B) \oplus_s (A \otimes_s C)$, as required. ∎

Proof (Proposition 8).

By definition of the kernel, $a \preceq b$ if and only if $Kernel_{\preceq}(\{a, b\}) = \{a\}$. The definition of $\oplus_s$ is $A \oplus_s B = Kernel_{\preceq}(A \cup B)$. Hence $\{a\} \oplus_s \{b\} = \{a\}$ iff $Kernel_{\preceq}(\{a, b\}) = \{a\}$. Therefore, $a \preceq b$ iff $\{a\} \oplus_s \{b\} = \{a\}$.

■

Proof (Proposition 9).

Directly follows from Proposition 7 ($\langle \mathbf{L}, \otimes_s, \oplus_s \rangle$ is a semiring) and the proof of Proposition 4 (Axioms A1, A2, A3 are satisfied for valuation structures generated by semirings). ■

Proof (Proposition 10).

Consider the original preference network $\mathcal{C}$ built on $\langle L, \preceq, \otimes \rangle$, and its set-based encoding $\mathbf{C}$ built on $\mathbf{L}$, i.e., $\mathbf{C}$ is the set of valuations $\mathbf{c}$ with $\mathbf{c}(d) = \{c(d)\}$ for any $c \in \mathcal{C}$. $scores(\mathbf{C})$ is the set of scores $e \in \mathbf{L}$ that are reached by some d in the set-based encoding of the original problem. Notice that, because the original scores given by the $\mathbf{c}$ in $\mathcal{C}$ are singletons for L, each e in $scores(\mathbf{C})$ is a singleton (this follows from the definition of $\otimes_s$ in $\mathbf{L}$). Moreover, $\{a\}$ is in $scores(\mathbf{C})$ iff $a \in scores(\mathcal{C})$. Hence, $scores(\mathcal{C}) = \bigcup_{e \in scores(\mathbf{C})} e$.

Now, recall that because $\oplus_s$ is associative and commutative $(\boxtimes_{\mathbf{c} \in \mathbf{C}} \mathbf{c})^{\downarrow\emptyset} = \bigoplus_{s\ e \in scores(\mathbf{C})} e$. Hence, $(\boxtimes_{\mathbf{c} \in \mathbf{C}}\ c)^{\downarrow\emptyset} = Kernel_{\preceq}(\bigcup_{e \in scores(\mathbf{C})} e) = Kernel_{\preceq}(scores(\mathcal{C}))$. ■

Proof (Proposition 11).

Let n be the number of variables, m the number of functions and $d = \max_{1 \leq i \leq n}\{|D_i|\}$ be the maximum domain size among the variables in the preference network.

The number of tuples of any new function computed by variable elimination is bounded by d^{tw}. The size of any element in $\mathbf{L}$ is bounded by $w(\preceq)$. Therefore, the size of any function is bounded by $w(\preceq) \times d^{tw}$. Since the number of new functions generated is bounded by the number of variables n, the space complexity clearly holds.

Let $A, B \in \mathbf{L}$. The time complexity of computing $A \otimes_s B$ is bounded by $w(\preceq)^2$. Therefore, the combination of two functions $\mathbf{c_i}, \mathbf{c_j} \in \mathbf{C}$ is bounded by $w(\preceq)^2 \times |D_{scope(\mathbf{c_i}) \cup scope(\mathbf{c_j})}|$. When eliminating variable x_i, there are at most $(m_i + deg_i - 1)$ functions to combine, where m_i is the number of original functions mentioning variable x_i and deg_i is the number of new functions resulting from the elimination of a previous variable and mentioning variable x_i. The combined arity of those functions is bounded by $tw + 1$. Therefore, the time complexity to eliminate variable x_i is bounded by $(m_i + deg_i - 1) \times d^{tw+1}$. The time complexity of variable elimination is bounded by $\sum_{i=1}^{n} (m_i + deg_i - 1) \times d^{tw+1}$. Since $\sum_{i=1}^{n} m_i = m$ and $\sum_{i=1}^{n} deg_i = n$, then the time complexity holds.

■

Proving Theorem 2

We state and prove a number of auxiliary results in order to prove this theorem.

Definition 9 (ordered group) *Let us define (for the purposes of this paper) that $\langle G, \otimes, \mathbf{1}, P\rangle$ is ordered group if*

- *G is a commutative group under $\otimes$ with identity element $\mathbf{1}$;*
- *$P \subseteq G$*
- *$P \ni \mathbf{1}$*
- *$a, b \in P \Rightarrow a \otimes b \in P$ (the elements of P are called "positive elements");*
- *$a, a^{-1} \in P \Rightarrow a = \mathbf{1}$.*

It is said to be a totally ordered group if for all $a \in G$ we either have $a \in P$ or $a^{-1} \in P$.

In the usual way, for group $(G, \otimes, \mathbf{1})$ and $a \in G$, we define $a^0 = \mathbf{1}$, and for $k = 1, 2, \ldots$ define a^k inductively by $a^k = a^{k-1} \otimes a$. For $k = -1, -2, \ldots$, define a^k to be $(a^{-1})^{-k}$.

Lemma 2. *Let $\langle G, \otimes, \mathbf{1}, P\rangle$ be an ordered group such that for all $a \in G$ with $a \neq \mathbf{1}$ and all $k > 0$ we have $a^k \neq \mathbf{1}$. Let a be some element of G with $a \neq \mathbf{1}$. Then either (i) for all $k > 0$, $a^k \notin P$, or (ii) for all $k > 0$, $(a^{-1})^k \notin P$.*

Proof. Suppose that neither (i) holds nor (ii) holds. Let k be minimal > 0 such that $a^k \in P$ and let l be minimal > 0 such that $a^{-l} \in P$. We have three cases:

- $k = l$: then $a^k \in P$ and $a^{-k} \in P$, i.e., $(a^{-1})^k \in P$, which implies by definition of an ordered group that $a^k = \mathbf{1}$, which implies $a = \mathbf{1}$, contradicting our hypothesis.
- $k > l$: Then $a^{k-l} \in P$, with $0 < k - l < k$, contradicting the definition of k.
- $k < l$: Then $(a^{-1})^{l-k} \in P$, with $0 < l - k < l$, contradicting the definition of l.

■

Lemma 3. *Let $\langle G, \otimes, \mathbf{1}, P\rangle$ be an ordered group, and let a be an element of G such that for all $k > 0$, $a^{-k} \notin P$. Then $\langle G, \otimes, \mathbf{1}, P'\rangle$ is an ordered group where $P' = \{a^k \otimes p \,:\, p \in P, k = 0, 1, 2, \ldots\}$.*

Proof. We have to show that P' satisfies the last three properties in Definition 9.

- $P' \ni \mathbf{1}$, by setting $k = 0$ and $p = \mathbf{1}$.
- Suppose P' contains two elements, which we can write as $a^k \otimes p$ and $a^l \otimes q$ for some $p, q \in P$ and $k, l \geq 0$. Then $(a^k \otimes p) \otimes (a^l \otimes q)$ equals $a^{k+l} \otimes (p \otimes q)$ which is an element of P' since $p \otimes q \in P$.
- Let $b = a^k \otimes p$ be an element of P' such that $b, b^{-1} \in P'$. Since $b^{-1} \in P'$ we can write b^{-1} as $a^l \otimes q$ for some $q \in P$ and $l \geq 0$. We can also write b^{-1} as $a^{-k} \otimes p^{-1}$. Hence $a^l \otimes q = a^{-k} \otimes p^{-1}$ and so $(a^{-1})^{k+l} = p \otimes q \in P$, which implies, by the assumed condition on a, that $k + l = 0$ and so $k = l = 0$. Thus $p \otimes q = \mathbf{1}$ and so $q = p^{-1}$ and therefore, $p, p^{-1} \in P$ which implies that $p = \mathbf{1}$. We have shown that $b = \mathbf{1}$, as required.

■

Lemma 4. *Let $\langle G, \otimes, \mathbf{1}, P\rangle$ be an ordered group which is not a totally ordered group such that for all $a \in G$ with $a \neq \mathbf{1}$ and all $k > 0$ we have $a^k \neq \mathbf{1}$. Then there exists a strict superset P' of P such that $\langle G, \otimes, \mathbf{1}, P'\rangle$ is an ordered group.*

Proof. Since $\langle G, \otimes, \mathbf{1}, P\rangle$ is not a totally ordered group there exists some $a \in G$ with $a \notin P$ and $a^{-1} \notin P$. $a \neq \mathbf{1}$, so, by Lemma 2, either (i) for all $k > 0$, $a^k \notin P$, or (ii) for all $k > 0$, $(a^{-1})^k \notin P$. First, suppose (ii) holds. Then by Lemma 3, $\langle G, \otimes, \mathbf{1}, P'\rangle$ is an ordered group where $P' = \{a^k \otimes p : p \in P, k = 0, 1, 2, \ldots\}$, which is a strict superset of P since $P' \ni a$. Case (i) is the same, except replacing a by a^{-1}. ■

Lemma 5. *Let $\langle G, \otimes, \mathbf{1}, P\rangle$ be an ordered group, and let $\mathcal{P}$ be the set of supersets P' of P such that $\langle G, \otimes, \mathbf{1}, P'\rangle$ is an ordered group. Let $\mathcal{S}$ be a non-empty subset of $\mathcal{P}$ which is totally ordered by set inclusion, and let $\bigcup\mathcal{S}$ be the union of the sets in $\mathcal{S}$. Then $\langle G, \otimes, \mathbf{1}, \bigcup\mathcal{S}\rangle$ is an ordered group.*

Proof. We just have to show that $\bigcup\mathcal{S}$ satisfies the three last properties in Definition 9.

- Clearly $\bigcup\mathcal{S} \ni \mathbf{1}$ since any element of $\mathcal{S}$ contains $\mathbf{1}$.
- Suppose that $a, b \in \bigcup\mathcal{S}$. Then there exists $P, P' \in \mathcal{S}$ with $a \in P$ and $b \in P'$, and so $a, b \in P \cup P'$. Now, since $\mathcal{S}$ is totally ordered by set inclusion, either $P \subseteq P'$ or $P' \subseteq P$, and so $P \cup P'$ equals either P or P', and hence $\langle G, \otimes, \mathbf{1}, P \cup P'\rangle$ is an ordered group; this implies that $a \otimes b \in P \cup P'$ and so $a \otimes b \in \bigcup\mathcal{S}$.
- Suppose that $a, a^{-1} \in \bigcup\mathcal{S}$. Applying the previous argument with $b = a^{-1}$ implies that a, a^{-1} are both elements of some set in $\mathcal{S}$, and so $a = \mathbf{1}$, by the definition of an ordered group.

■

Proposition 12 *Let $\langle G, \otimes, \mathbf{1}, P\rangle$ be an ordered group such that for all $a \in G$ with $a \neq \mathbf{1}$ and all $k > 0$ we have $a^k \neq \mathbf{1}$. Then there exists some superset P^* of P such that $\langle G, \otimes, \mathbf{1}, P^*\rangle$ is a totally ordered group.*

Proof. Let $\mathcal{P}$ be the set of all supersets P' of P such that $\langle G, \otimes, \mathbf{1}, P'\rangle$ is an ordered group. $\mathcal{P}$ is partially ordered by set inclusion. Lemma 5 shows that every totally ordered subset of $\mathcal{P}$ has an upper bound (in $\mathcal{P}$), so by Zorn's Lemma, $\mathcal{P}$ has at least one maximal element; call this element P^*. We will show that $\langle G, \otimes, \mathbf{1}, P^*\rangle$ is a totally ordered group. Suppose otherwise. Then by Lemma 4, there exists a strict superset Q of P^* such that $\langle G, \otimes, \mathbf{1}, Q\rangle$ is an ordered group, and so $Q \in \mathcal{P}$, which contradicts the maximality of P^*. ■

Proposition 13 *Let $(G, \otimes)$ be a commutative group with operation $\otimes$ and identity element $\mathbf{1}$, such that for all $a \in G$ with $a \neq \mathbf{1}$ and all $k > 0$ we have $a^k \neq \mathbf{1}$. Let $\preceq$ be a partial order on G satisfying monotonicity with respect to $\otimes$ (i.e., for all $a, b, c \in G$, $a \preceq b$ implies $a \otimes c \preceq b \otimes c$). Then there exists a total order $\preceq^*$ extending $\preceq$ with $\preceq^*$ satisfying monotonicity with respect to $\otimes$.*

Proof. Define $P = \{a \in G : a \succeq \mathbf{1}\}$. Then $\langle G, \otimes, \mathbf{1}, P\rangle$ is an ordered group:

- $P \ni \mathbf{1}$ since $\mathbf{1} \succeq \mathbf{1}$.
- Suppose that $a, b \in P$. Then $a, b \succeq \mathbf{1}$, so by monotonicity $a \otimes b \succeq \mathbf{1} \otimes b = b \succeq \mathbf{1}$ and hence, $a \otimes b \in P$.
- Suppose that $a, a^{-1} \in P$. Then $a^{-1} \succeq \mathbf{1}$, so by monotonicity, $\mathbf{1} = a \otimes a^{-1} \succeq a \otimes \mathbf{1} = a$ implying $\mathbf{1} \succeq a \succeq \mathbf{1}$, and hence $a = \mathbf{1}$, since $\succeq$ is a partial order.

Then, by Proposition 12, there exists some superset P^* of P such that $\langle G, \otimes, \mathbf{1}, P^*\rangle$ is a totally ordered group. Define relation $\preceq^*$ on G by $a \preceq^* b$ if and only if $b \otimes a^{-1} \in P^*$. We will show that $\preceq^*$ is a total order extending $\preceq$ satisfying monotonicity.

- Suppose that $a \preceq b$. Then, by monotonicity, $\mathbf{1} \preceq b \otimes a^{-1}$ so $b \otimes a^{-1} \in P$. Hence $b \otimes a^{-1} \in P^*$ which implies that $a \preceq^* b$. We have shown that $\preceq^*$ extends $\preceq$.
- Monotonicity: Suppose $a \preceq^* b$. Then $b \otimes a^{-1} \in P^*$. Now, $(b \otimes c) \otimes (a \otimes c)^{-1} = b \otimes c \otimes a^{-1} \otimes c^{-1} = b \otimes a^{-1}$ and so $(b \otimes c) \otimes (a \otimes c)^{-1} \in P^*$. Thus $a \otimes c \preceq^* b \otimes c$.
- Reflexivity: $a \preceq^* a$ since $a \otimes a^{-1} = \mathbf{1} \in P^*$.
- Completeness: Let $a, b \in G$. Since $\langle G, \otimes, \mathbf{1}, P^*\rangle$ is a totally ordered group, either (i) $b \otimes a^{-1} \in P^*$ and so $a \preceq^* b$, or (ii) $(b \otimes a^{-1})^{-1} \in P^*$, i.e., $a \otimes b^{-1} \in P^*$ and so $b \preceq^* a$.
- Transitivity: Suppose that $a \preceq^* b$ and $b \preceq^* c$. Then $b \otimes a^{-1} \in P^*$ and $c \otimes b^{-1} \in P^*$ so $b \otimes a^{-1} \otimes c \otimes b^{-1} \in P^*$ so $c \otimes a^{-1} \in P^*$ and therefore $a \preceq^* c$.
- Anti-symmetric: Suppose that $a \preceq^* b$ and $b \preceq^* a$. This implies that $b \otimes a^{-1} \in P^*$ and $a \otimes b^{-1} \in P^*$. Since $a \otimes b^{-1} = (b \otimes a^{-1})^{-1}$ this implies that $a \otimes b^{-1} = \mathbf{1}$ and so $a = b$.

■

Lemma 6. *Suppose that $\otimes$ is a commutative and associative operation over a set L, which is monotonic with respect to a partial order $\preceq$ and also satisfies the property $a \otimes c \preceq b \otimes c \Rightarrow a \preceq b$. Define $H = L \times L$. Define operation $\otimes$ on H by $(a, b) \otimes (c, d) = (a \otimes c, b \otimes d)$. Define relation $\preceq$ on H by $(a, b) \preceq (c, d) \iff a \otimes d \preceq b \otimes c$. Define relation $\equiv$ on H to be the symmetric part of $\preceq$, so that $(a, b) \equiv (c, d)$ if and only if $(a, b) \preceq (c, d)$ and $(a, b) \succeq (c, d)$. Then:*

- *$\preceq$ on H is reflexive and transitive;*
- *$\equiv$ is an equivalence relation;*
- *$\otimes$ on H is commutative and associative.*
- *$(a, b) \equiv (a', b')$ if and only if $a \otimes b' = a' \otimes b$.*

– *if* $(a, b) \equiv (a', b')$ *and* $(c, d) \equiv (c', d')$ *then* $(a \otimes c, b \otimes d) \equiv (a' \otimes c', b' \otimes d')$.
– $(a, b) \equiv (c, d)$ *if and only if* $(b, a) \equiv (d, c)$.

Proof.

– By commutativity of $\otimes$ on L and reflexivity of $\preceq$ on L we have $a \otimes b \preceq b \otimes a$ which implies that $(a, b) \preceq (a, b)$, proving reflexivity of $\preceq$ on H.
– $(a, b) \preceq (c, d)$ and $(c, d) \preceq (e, f)$ implies $a \otimes d \preceq b \otimes c$ and $c \otimes f \preceq d \otimes e$. Hence $a \otimes d \otimes f \preceq b \otimes c \otimes f$ and $b \otimes c \otimes f \preceq b \otimes d \otimes e$, and so $a \otimes d \otimes f \preceq b \otimes d \otimes e$, which implies that $a \otimes f \preceq b \otimes e$ and so $(a, b) \preceq (e, f)$, proving transitivity.
– $\preceq$ on H is reflexive and transitive implies that $\equiv$ is reflexive, symmetric and transitive and hence an equivalence relation;
– $\otimes$ on H is clearly commutative and associative, since $\otimes$ on L is commutative and associative;
– $(a, b) \equiv (a', b')$ if and only if $(a, b) \preceq (a', b')$ and $(a', b') \preceq (a, b)$ which is if and only if $a \otimes b' \preceq a' \otimes b$ and $a' \otimes b \preceq a \otimes b'$, which is if and only if $a \otimes b' = a' \otimes b$.
– Suppose $(a, b) \equiv (a', b')$ and $(c, d) \equiv (c', d')$. Then, by the last part, $a \otimes b' = a' \otimes b$ and $c \otimes d' = c' \otimes d$. Hence $a \otimes b' \otimes c \otimes d' = a' \otimes b \otimes c' \otimes d$, and so $(a \otimes c) \otimes (b' \otimes d') = (a' \otimes c') \otimes (b \otimes d)$, which implies $(a \otimes c, b \otimes d) \equiv (a' \otimes c', b' \otimes d')$.
– $(b, a) \equiv (d, c)$ if and only if $b \otimes c = a \otimes d$ which is if and only if $(a, b) \equiv (c, d)$.

∎

Lemma 7. *Suppose that $\otimes$ is a commutative and associative operation over a set L, which is monotonic with respect to a partial order $\preceq$ and also satisfies the property $a \otimes c \preceq b \otimes c \Rightarrow a \preceq b$. Define $H = L \times L$ and define $\equiv$ and $\otimes$ on H as in Lemma 6. Define $G = H/\equiv$, i.e., the set of equivalence classes of H under $\equiv$. $\preceq$ on H induces relation $\preceq$ on G, which is partial order. For $(a, b) \in H$ we write $[(a, b)] \in G$ for the equivalence class containing (a, b). Define $\otimes$ on G by $[(a, b)] \otimes [(c, d)] = [(a \otimes c, b \otimes d)]$, (this is well-defined by Lemma 6). Then*

– *$(G, \otimes)$ is a commutative group with identity element* $\mathbf{1}$ *being the equivalence class consisting of all elements of the form (a, a);*
– *$\otimes$ on G is monotonic over $\preceq$, and we also have for $e, f, g \in G$, that $e \otimes g \preceq f \otimes g \Rightarrow e \preceq f$.*

Proof. $\preceq$ on G is clearly a partial order since $\preceq$ on H is a reflexive and transitive relation, and $\equiv$, the symmetric part of $\preceq$, is an equivalence relation, and $\preceq$ on G is $\preceq$ on H factored by equivalence $\equiv$. (It is anti-symmetric since if $h, h' \in H$ and $[h]$ is the equivalence class containing h then $[h] \preceq [h'] \preceq [h]$ implies $h \preceq h' \preceq h$, and so $h \equiv h'$ and therefore $[h] = [h']$.) $\otimes$ on G is commutative and associative because $\otimes$ on H is commutative and associative. For any $c, d \in L$, $(c, c) \equiv (d, d)$. For any $[(a, b)] \in G$, $[(a, b)] \otimes [(c, c)] = [(a \otimes c, b \otimes c)] = [(a, b)]$ since $(a \otimes c, b \otimes c) \equiv (a, b)$ (e.g., using Lemma 6). Hence $[(c, c)]$ is an identity element, and so is the unique identity element. Also, if $(a, b) \equiv (c, c)$ then $a \otimes c = b \otimes c$, so $a \preceq b$ and

$b \preceq a$ and therefore, $a = b$; hence the equivalence class of (c, c) consists of exactly those elements of the form (a, a) for some a. Define $[(a, b)]^{-1} = [(b, a)]$ (this is well-defined by Lemma 6). $[(a, b)] \otimes [(b, a)] = [(a \otimes b, a \otimes b)]$, which is the identity element. Hence any element has an inverse, and so $(G, \otimes)$ is a commutative group with identity element $\mathbf{1}$ being the equivalence class consisting of all elements of the form (a, a).

Now, $[(a, b)] \preceq [(c, d)]$ holds if and only if $a \otimes d \preceq b \otimes c$. $[(a, b)] \otimes [(e \otimes f)] \preceq [(c, d)] \otimes [(e \otimes f)]$ holds if and only if $[(a \otimes e, b \otimes f)] \preceq [(c \otimes e, d \otimes f)]$ which is if and only if $a \otimes e \otimes d \otimes f \preceq b \otimes f \otimes c \otimes e$, which is if and only if $a \otimes d \preceq b \otimes c$. Therefore, $[(a, b)] \preceq [(c, d)]$ holds if and only if $[(a, b)] \otimes [(e \otimes f)] \preceq [(c, d)] \otimes [(e \otimes f)]$. ■

The following result is a rewriting of Theorem 2.

Theorem 3 *Suppose that $\otimes$ is a commutative and associative operation over a set L with a neutral element $\mathbf{1}$, where $\otimes$ is monotonic with respect to a partial order $\preceq$ and also satisfies the property $a \otimes c \preceq b \otimes c \Rightarrow a \preceq b$ for all $a, b, c \in L$. Suppose also that for all $a, b \in L$ with $a \neq b$ and all $k > 0$ we have $a^k \neq b^k$. Then there exists a total order $\preceq'$ on L extending $\preceq$ and such that for all $a, b, c \in L$, $a \otimes c \preceq' b \otimes c \iff a \preceq' b$.*

Proof. Consider group $(G, \otimes)$ with identity element $\mathbf{1}_G$, and partial order relation $\preceq$ on G, as defined in Lemma 7. Consider any $g \in G$, and any $k > 0$ such that $g^k = \mathbf{1}_G$. g represents an $\equiv$-equivalence class in H. Let (a, b) be an element in this equivalence class. We have $(a, b)^k \equiv (a, a)$, i.e., $(a^k, b^k) \equiv (a, a)$. By definition of $\equiv$ we have $a^k \otimes a \preceq b^k \otimes a$, and $a^k \otimes a \succeq b^k \otimes a$, and so $a^k \preceq b^k$ and $a^k \succeq b^k$, and so $a^k = b^k$. By our hypothesis this implies that $a = b$, which implies that $(a, b) = (a, a)$ which implies that $g = \mathbf{1}_G$. This means that the conditions of Proposition 13 are satisfied.

Applying Proposition 13, there exists a total order $\preceq^*$ extending $\preceq$ on G with $\preceq^*$ satisfying monotonicity with respect to $\otimes$. Define relation $\preceq'$ on L as follows: for $a, b \in L$, $a \preceq' b \iff [(a, \mathbf{1})] \preceq^* [(b, \mathbf{1})]$, where $[(c, d)]$ means the equivalence class containing (c, d). We can show that $\preceq'$ is a total order on L:

- $\preceq'$ is reflexive since $\preceq^*$ is reflexive;
- if $a \preceq' b \preceq' c$ then $[(a, \mathbf{1})] \preceq^* [(b, \mathbf{1})] \preceq^* [(c, \mathbf{1})]$, and so $[(a, \mathbf{1})] \preceq^* [(c, \mathbf{1})]$ by transitivity of $\preceq^*$, and hence $a \preceq' c$, proving transitivity of $\preceq'$;
- if $a \preceq' b \preceq' a$ then $[(a, \mathbf{1})] \preceq^* [(b, \mathbf{1})] \preceq^* [(a, \mathbf{1})]$, which implies that $[(a, \mathbf{1})] = [(b, \mathbf{1})]$, and so $(a, \mathbf{1}) \equiv (b, \mathbf{1})$ and hence $a \preceq b$ and $b \preceq a$, which implies $a = b$. This shows that $\preceq'$ is anti-symmetric.
- Consider any $a, b \in L$. By completeness of $\preceq^*$ we either have $[(a, \mathbf{1})] \preceq^* [(b, \mathbf{1})]$ or $[(a, \mathbf{1})] \succeq^* [(b, \mathbf{1})]$ and hence either $a \preceq' b$ or $a \succeq' b$ proving that $\preceq'$ is complete.

We also have that $\preceq'$ extends $\preceq$: suppose $a \preceq b$; then $[(a, \mathbf{1})] \preceq [(b, \mathbf{1})]$ and so $[(a, \mathbf{1})] \preceq^* [(b, \mathbf{1})]$, and hence $a \preceq' b$.

For any $a, b, c \in L$, we have $a \otimes c \preceq' b \otimes c \iff [(a \otimes c, \mathbf{1})] \preceq^* [(b \otimes c, \mathbf{1})] \iff [(a, \mathbf{1})] \otimes [(c, \mathbf{1})] \preceq^* [(b, \mathbf{1})] \otimes [(c, \mathbf{1})] \iff [(a, \mathbf{1})] \otimes [(c, \mathbf{1})] \otimes [(\mathbf{1}, c)] \preceq^* [(b, \mathbf{1})] \otimes$

$[(c, \mathbf{1})] \otimes [(\mathbf{1}, c)]$ if and only if $[(a, \mathbf{1})] \preceq^* [(b, \mathbf{1})]$ (since e.g., $[(a \otimes c, c)] = [(a, \mathbf{1})]$). This is if and only if $a \preceq' b$, completing the proof. ∎

Corollary 1. *Suppose that $\otimes$ is a commutative and associative operation over a set L with a neutral element $\mathbf{1}$, where $\otimes$ is monotonic with respect to a partial order $\preceq$ and also satisfies the property $a \otimes c \preceq b \otimes c \Rightarrow a \preceq b$ for all $a, b, c \in L$ with $c \neq \top, \bot$, where $\top$ and $\bot$ satisfy the properties:*

- *for all $a \in L$, $\bot \preceq a \preceq \top$*
- *for all $a \in L - \{\top\}$, $a \otimes \bot = \bot$*
- *for all $a \in L - \{\bot\}$, $a \otimes \top = \top$*
- *for all $a, b \in L - \{\bot, \top\}$, $a \otimes b \neq \bot, \top$.*

Suppose also that for all $a, b \in L$ with $a \neq b$ and all $k > 0$ we have $a^k \neq b^k$. Then there exists a total order $\preceq'$ on L extending $\preceq$ where $\otimes$ is monotonic with respect to $\preceq'$, and $a \otimes c \preceq' b \otimes c \Rightarrow a \preceq' b$ for all $a, b, c \in L$ with $c \neq \top, \bot$.

Proof. We must have $\top \neq \mathbf{1}$ and $\bot \neq \mathbf{1}$. Then we can apply Theorem 2 to $L - \{\bot, \top\}$ with $\otimes$ and $\preceq$ restricted to this set. So there exists a total order $\preceq'$ on $L - \{\bot, \top\}$ extending $\preceq$ on $L - \{\bot, \top\}$ and such that for all $a, b, c \in L - \{\bot, \top\}$, $a \otimes c \preceq' b \otimes c \iff a \preceq' b$. We extend $\preceq'$ to L by for all $a \in L$, $\bot \preceq a \preceq \top$. $\preceq'$ on L is clearly a total order which extends $\preceq$.

To prove monotonicity: assume monotonicity fails with respect to $\preceq'$, so that there exists $a, b, c \in L$ and $a \preceq' b$ such that $a \otimes c \not\preceq' b \otimes c$, so $a \otimes c \neq b \otimes c$ and $a \otimes c \succeq' b \otimes c$. It must be the case that $a \not\preceq b$ (since $a \preceq b$ implies $a \otimes c \preceq b \otimes c$ and so $a \otimes c \preceq' b \otimes c$). Thus $a, b \in L - \{\bot, \top\}$. By monotonicity of $\preceq'$ on $L - \{\bot, \top\}$ we must have $c \in \{\bot, \top\}$. But then $a \otimes c = b \otimes c$, which is a contradiction.

To prove the final property by contradiction, suppose that $a \otimes c \preceq' b \otimes c$ but $a \not\preceq' b$ for some $a, b, c \in L$ with $c \neq \top, \bot$. Since $\preceq'$ is a total order this implies $b \prec' a$. We have $a \neq \top$, since if $a = \top$ then $a \otimes c = \top$ which contradicts $a \otimes c \preceq' b \otimes c$. Similarly, $b \neq \bot$. $b \prec' a$ then implies that $b \neq \top$ and $a \neq \bot$. Then $a, b \in L - \{\bot, \top\}$. We then have $a \otimes c \preceq' b \otimes c \Rightarrow a \preceq' b$ which is the contradiction required. ∎

Acknowledgements

The work of the third author has been supported by the Science Foundation Ireland under Grant No. 08/PI/I1912.

References

1. S. Benferhat, S. Lagrue, and O. Papini. Revision of partially ordered information: Axiomatization, semantics and iteration. In *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, pages 376–381, 2005.

2. S. Bistarelli, U. Montanari, and F. Rossi. Constraint solving over semirings. In *IJCAI'95*, pages 624–630, 1995.
3. S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based constraint satisfaction and optimization. *J. ACM*, 44(2):201–236, 1997.
4. C. Boutilier, R. Brafman, C. Domshlak, H. Hoos, and D. Poole. CP-nets: a tool for representing and reasoning with conditional ceteris paribus statements. *JAIR*, 21:135–191, 2004.
5. G. Brewka. Preferred subtheories: An extended logical framework for default reasoning. In *IJCAI'89*, pages 1043–1048, 1989.
6. D. Dubois and H. Fargier. Qualitative decision making with bipolar information. In *KR'06*, pages 175–186, 2006.
7. D. Dubois, J. Lang, and H. Prade. Timed possibilistic logic. *Fundam. Inform.*, 15(3-4):211–234, 1991.
8. Jean-Philippe Dubus, Christophe Gonzales, and Patrice Perny. Multiobjective optimization using GAI models. In *IJCAI*, pages 1902–1907, 2009.
9. H. Fargier and N. Wilson. Local computation schemes with partially ordered preferences. In *ECSQARU*, pages 34–45, 2009.
10. E. Freuder and R. Wallace. Partial constraint satisfaction. *Artificial Intelligence*, 58(1-3):21–70, 1992.
11. N. Friedman and J. Halpern. Plausibility measures and default reasoning. In *AAAI'96*, pages 1297–1304, 1996.
12. E. Giunchiglia and M. Maratea. Planning as satisfiability with preferences. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence*, pages 987–992, 2007.
13. M.I. Henig. The shortest path problem with two objective functions. *EJOR*, 25:281–291, 1985.
14. J. Kohlas. *Information Algebras: Generic Structures for Inference*. Springer-Verlag, 2003.
15. J. Kohlas and N. Wilson. Semiring induced valuation algebras: Exact and approximate local computation algorithms. *Artificial Intelligence*, 172:1360–1399, 2008.
16. S. Lindström and W. Rabinowicz. Epistemic entrenchment with incomparabilities and relational belief revision. In *The Logic of Theory Change, Workshop, Konstanz, FRG, October 13-15, 1989, Proceedings*, pages 93–126, 1989.
17. Christos H. Papadimitriou and Mihalis Yannakakis. On the approximability of trade-offs and optimal access of web sources (extended abstract). In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 86–92, 2000.
18. E. Rollon. *Multi-Objective Optimization for Graphical Models*. PhD thesis, Universitat Politècnica de Catalunya, Barcelona, Spain, 2008.
19. E. Rollon and J. Larrosa. Bucket elimination for multiobjective optimization problems. *J. Heuristics*, 12(4-5):307–328, 2006.
20. B. Roy. The outranking approach and the foundations of ELECTRE methods. *Theory and Decision*, 31(1):49–73, 1991.
21. T. Schiex, H. Fargier, and G. Verfaillie. Valued constraint satisfaction problems: Hard and easy problems. In *IJCAI'95*, pages 631–637, Montreal, 1995.
22. P. P. Shenoy. Valuation-based systems for discrete optimisation. In *UAI'90*, pages 385–400, 1990.
23. P. P. Shenoy and G. Shafer. Axioms for probability and belief-function proagation. In *UAI*, pages 169–198, 1988.
24. N. Wilson. An order of magnitude calculus. In *UAI'95*, pages 548–555, 1995.
25. N. Wilson. A logic of soft constraints based on partially ordered preferences. *Journal of Heuristics*, 12(4-5):241–262, 2006.