Soft Constraints with Partially Ordered Preferences[1]

Nic Wilson[2]

Abstract. This paper constructs a logic of soft constraints where the set of degrees of preference forms a partially ordered set. When the partially ordered set is a distributive lattice, this reduces to the idempotent semiring-based CSP approach, and the lattice operations can be used to define a sound and complete proof theory. For the general case, it is shown how sound and complete deduction can be performed by using a particular embedding of a partially ordered set in a distributive lattice.

1 INTRODUCTION

Representing and reasoning with an agent's preferences is important in many applications of constraints formalisms. Preferences are often most naturally only partially ordered, reflecting an agent being unable or unwilling to order certain choices, or wishing to delay making such an ordering decision. Most soft constraints formalisms assume a total order on the degrees of preference. On the other hand, the elegant and general semiring-based CSP framework [1, 2] does allow a partially ordered set of preference degrees, but this partially ordered set must form a distributive lattice; whilst this is convenient computationally, it restricts the representational power. This paper constructs a logic of soft constraints where it is only assumed that the set of preference degrees is a partially ordered set, with a maximum element 1 and a minimum element 0. A soft constraint assigns a preference degree to a tuple, which is interpreted as an upper bound for the overall preference degree of the tuple. When the partially ordered set is a distributive lattice, this reduces to the idempotent semiring-based CSP approach, and the lattice operations can be used to define a sound and complete proof theory, as shown in section 2.2. This case can also be viewed as a lattice-valued possibilistic logic [3], as shown in [6]. In section 2.3 the generally partially ordered case is considered. It is shown how a particular embedding of a partially ordered set in a distributive lattice allows sound and complete deduction to be performed in the general case, by using deduction in the distributive lattice case.

This paper is a condensed version of [7], which is based on [6].

2 A LOGIC OF SOFT CONSTRAINTS

Let V be a finite set of variables, where each variable $X \in V$ has finite domain $\underline{X}$. For $U \subseteq V$, define $\underline{U}$ to be the set of possible assignments to variables U, that is, $\prod_{X \in U} \underline{X}$. A complete tuple x is an element of $\underline{V}$. For $U \subseteq V$ let $x^{\downarrow U}$ be the projection of x to variables U.

The intention is to produce a formalism that allows degrees of preference (or satisfaction, or adequacy) for partial tuples. So we choose a finite partially ordered set $\mathcal{A} = (A, \preceq, 0, 1)$ to represent these degrees, where A contains a maximum element 1 and a minimum element 0. Define an $\mathcal{A}$-constraint c to be a function from $\underline{V_c}$ to A, for some set of variables $V_c \subseteq V$. A value of 0 means that the tuple is least preferred, a value of 1 expresses no information.

2.1 Semantics

We would like to say what $\mathcal{A}$-constraints d can be deduced from a set of $\mathcal{A}$-constraints C. We imagine that there exists some (unknown) function from $\underline{V}$ to A which gives the 'true' degree of preference (of e.g., a user) of each complete tuple $x \in \underline{V}$. Constraint c is understood as giving upper bounds on the degrees of preference; we say, for function $M : \underline{V} \to A$, that M *satisfies* c (written: $M \models c$) if and only if for all $x \in \underline{V}$, $M(x) \preceq c(x^{\downarrow V_c})$. For set of $\mathcal{A}$-constraints C and $\mathcal{A}$-constraint d we define $C \models d$ if and only if every M satisfying (every constraint in) C also satisfies d. Soft constraints C express information about the preferences (of e.g., a user), and so such a d expresses derived preference information, which may be thought of as a property of the (user's) preferences.

The semantic definition is not very helpful for computing the consequences d of C; we need some more computationally useful characterisation. To do this end we consider a special case first.

2.2 The case when $\mathcal{A}$ is a distributive lattice

We first look at this special case of a partial order. For finite A, $(A, 0, 1, \preceq)$ is a distributive lattice if and only if any $\alpha, \beta \in A$ have a greatest lower bound $\alpha \wedge \beta$ in A (so that $\gamma \preceq \alpha, \beta$ implies $\gamma \preceq \alpha \wedge \beta \preceq \alpha, \beta$), and a least upper bound $\alpha \vee \beta$ in A (so that $\alpha, \beta \preceq \gamma$ implies $\alpha, \beta \preceq \alpha \vee \beta \preceq \gamma$), which satisfy the distributivity property: for all $\alpha, \beta, \gamma \in A$, $\gamma \wedge (\alpha \vee \beta) = (\gamma \wedge \alpha) \vee (\gamma \wedge \beta)$.

The lattice properties enable us to define combination and projection of $\mathcal{A}$-constraints. Let $c : V_c \to A$ and $d : V_d \to A$ be two $\mathcal{A}$-constraints. Their combination $c \wedge d$ is the $\mathcal{A}$-constraint on variables $V_c \cup V_d$ given by, for $y \in \underline{V_c \cup V_d}$, $(c \wedge d)(y) = c(y^{\downarrow V_c}) \wedge d(y^{\downarrow V_d})$. For $U \subseteq V_c$, $c^{\downarrow U}$, the projection of c to U, is given by: for $u \in \underline{U}$, $c^{\downarrow U}(u) = \bigvee \{c(y) : y \in \underline{V_c}, y^{\downarrow U} = u\}$. 1_U is the constraint on variables U which is everywhere equal to 1: for all $u \in \underline{U}$, $1_U(u) = 1$.

A simple sound and complete proof theory can be defined using these operations. Define the proof theory by the axiom 1_V and inference rules:

— *From c and d deduce $c \wedge d$.*
— For each constraint c and $U \subseteq V_c$ the following inference rule: *From c deduce $c^{\downarrow U}$.*
— When $V_c = V_d$ and $c \preceq d$ (i.e., for all $y \in \underline{V_c}$, $c(y) \preceq d(y)$): *From c deduce d.*

---

[1] This work has received support from Science Foundation Ireland under Grant 00/PI.1/C075
[2] Cork Constraint Computation Centre, Department of Computer Science, University College Cork, Cork, Ireland, n.wilson@4c.ucc.ie

Theorem 1 (Soundness and Completeness) *$C \models d$ if and only if $(\bigwedge C \wedge 1_{V_d})^{\downarrow V_d} \preceq d$ if and only if d can be proved from C using the axiom and inference rules.*

An important derived inference rule is elimination of a variable: For variable $X \in V$, combine all constraints involving that variable and project to $U - \{X\}$, where U is the set of variables involved in the combination. In fact [6, 7] this can be used as a complete proof procedure, which is efficient if a good hypertree/join tree decomposition can be found [5, 4].

The formalism defined in 2.1 and 2.2 is strongly related to semiring-based CSPs (Bistarelli et al, 97) [1]; as we showed in [6], the notion of consequence defined above is the same as the natural notion of consequence in idempotent semiring-based CSPs. So we have a new and, in a certain sense, deeper, semantics for idempotent semiring-based CSPs. The previous semantics [2] treats the operations of multiplication $\wedge$ and addition $\vee$ as primitives, and defines the semantics in terms of them. The strength of this new semantics is that it assumes so little: essentially the whole system follows from saying that the constraints express upper bounds on preference degrees for complete assignments [7].

2.3 The General Partially Ordered Set Case

Here we consider the general case, where $\mathcal{A} = (A, 0, 1, \preceq)$ is an arbitrary partially ordered set that has a minimum element 0 and a maximum element 1. This is important because we cannot in general expect a set of partially ordered preference degrees to form a distributive lattice. Our approach is to embed the partially ordered set in a lattice of subsets in such a way that the ordering information is maintained, but without adding any extra ordering information. Then we can use the proof procedures for the subset lattice case to make deductions for this general case. The following proposition gives sufficient conditions on an embedding Q for this to work.

Proposition 1 *Let Θ be a finite set and let Q be a function from A to 2^Θ satisfying (i) $\alpha \preceq \beta \iff Q(\alpha) \subseteq Q(\beta)$, and (ii) for any $B \subseteq A$ and $\theta \in \bigcap_{\beta \in B} Q(\beta)$, there exists $\alpha \in A$ with $\theta \in Q(\alpha) \subseteq \bigcap_{\beta \in B} Q(\beta)$. For each $c : \underline{V_c} \to A$ define $c^Q : \underline{V_c} \to 2^\Theta$ to be c followed by Q, so that $c^Q(y) = Q(c(y))$.*

Let C be a set of $\mathcal{A}$-constraints, and define C^Q to be $\{c^Q : c \in C\}$. Then $C \models d$ if and only if $C^Q \models d^Q$.

It is easy to find an embedding with the appropriate properties. In particular we can define Θ to be A and $Q(\alpha)$ to be $\{\beta \in A : \beta \preceq \alpha\}$ (see below). However, if A is large, working with subsets of A can be computationally expensive. We give a way of constructing an embedding which can lead to much smaller Θ than A. We will consider embeddings of a particular form. Let A' be a subset of A. For $\alpha \in A$ let $Q_{A'}(\alpha)$ be the set $\{\alpha' \in A' : \alpha' \preceq \alpha\}$. It can easily be seen that for any A', the mapping $\alpha \mapsto Q_{A'}(\alpha)$ satisfies (ii) above (using $\alpha = \theta$), and half of (i): if $\alpha \preceq \beta$ then $Q_{A'}(\alpha) \subseteq Q_{A'}(\beta)$, (by transitivity of $\preceq$). Setting $A' = A - \{0\}$ we get the other half of (i): if $Q_{A'}(\alpha) \subseteq Q_{A'}(\beta)$ then $\alpha \preceq \beta$. However, we can very often find a much smaller A' that still satisfies both conditions (i) and (ii) of the proposition, as shown below.

Construction of a particular A'

For $B \subseteq A$ define relations $\preceq_B$ by $\alpha \preceq_B \beta$ if and only if $\gamma \preceq \beta$ for all $\gamma \in B$ such that $\gamma \preceq \alpha$, i.e., if and only if $Q_B(\alpha) \subseteq Q_B(\beta)$. These relations contain $\preceq$, and are monotonic (decreasing) with respect to B: if $B' \subseteq B$ then $\preceq_{B'} \supseteq \preceq_B \supseteq \preceq$, so that $\alpha \preceq \beta$ implies $\alpha \preceq_B \beta$, which implies $\alpha \preceq_{B'} \beta$.

Let $m = |A| - 1$. We list the elements of A in an order compatible with $\preceq$, starting with 0, so that $\alpha_0 = 0$, and if $\alpha_i \preceq \alpha_j$ then $i \leq j$. We build up A', element by element, with the final set A' being A_m.

Define $A_0 = \emptyset$, and for $i = 1, \ldots, m$, define Y_i, Z_i and A_i inductively as follows:

— set $Y_i = \{\alpha_i\}$ if there exists $k < i$ with $\alpha_i \preceq_{A_{i-1}} \alpha_k$; otherwise set $Y_i = \emptyset$;
— set Z_i to be the set of all α_j such that (a) $j < i$, (b) $\alpha_j \not\preceq \alpha_i$, and (c) $\alpha_j \preceq_{A_{i-1}} \alpha_i$;
— let $A_i = A_{i-1} \cup Y_i \cup Z_i$

(Adding Y_i ensures that $\alpha_i \not\preceq_{A_i} \alpha_k$ and hence $\alpha_i \not\preceq_{A'} \alpha_k$ for $k > i$; adding Z_i ensures that $\alpha_j \not\preceq_{A_i} \alpha_i$, and hence $\alpha_j \not\preceq_{A'} \alpha_i$, if $j < i$ and $\alpha_j \not\preceq \alpha_i$.)

Finally, we let $A' = A_m$, and define $Q = Q_{A'}$, i.e., for each $\alpha \in A$, $Q(\alpha) = \{\beta \in A' : \beta \preceq \alpha\}$.

Theorem 2 *With the above definition of A' and Q, the relations $\preceq$ and $\preceq_{A'}$ are the same, and for all $\alpha, \beta \in A$, $\alpha \preceq \beta$ if and only if $Q(\alpha) \subseteq Q(\beta)$. Furthermore, $C \models d$ if and only if $C^Q \models d^Q$.*

The purpose of the construction is to produce an embedding into not too large a set. In the worst case, when $\preceq$ is a total order, the set A' is just $A - \{0\}$, so $|A'| = |A| - 1$. The other extreme is when A is the lattice of all subsets of a set Ω. Then A' is the set of singleton subsets of Ω so $|A'| = |\Omega| = \log_2 |A|$.

This theorem gives us a sound and complete method for determining if inferences of the form $C \models d$ hold (for general $\mathcal{A}$-constraints): we construct A' and Q as defined above, convert the $\mathcal{A}$-constraints into $2^{\mathcal{A}'}$-constraints, and use the machinery provided for the distributive lattice case (Theorem 1, and in particular, the variable elimination inference rule) to determine if $C^Q \models d^Q$. This procedure will be efficient if the partially ordered set is not too large, given appropriate structure on the constraints' variables.

REFERENCES

[1] Bistarelli, S., Montanari, U., Rossi, F., Semiring-based Constraint Solving and Optimization, *Journal of the ACM*, vol. 44, no. 2, 201–236 (1997).

[2] Bistarelli, S., Montanari, U., Rossi, F., Semiring-Based Constraint Logic Programming: Syntax and Semantics, *ACM Transactions on Programming Languages and Systems*, ACM Press New York, USA, Pages: 1–29 Vol. 23, issue 1, (2001).

[3] Dubois, D., Lang, J., Prade, H., Possibilistic Logic, *Handbook of Logic in Artificial Intelligence and Logic Programming*, D. Gabbay, C. Hogger, J. Robinson (eds.), Vol. 3, 439–513, Oxford University Press (1994).

[4] Kohlas, J., and Shenoy, P., Computation in Valuation Algebras, *Volume 5: Algorithms for Uncertainty and Defeasible Reasoning* J. Kohlas, S. Moral (eds.), *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Kluwer Academic Publishers (2000).

[5] Shenoy, P. P., and Shafer, G., Axioms for Probability and Belief Function Propagation, in *Uncertainty in Artificial Intelligence 4*, R. D. Shachter, T. S. Levitt, L. N. Kanal, J. F. Lemmer (eds.), North Holland, also in *Readings in Uncertain Reasoning*, Shafer, G., and Pearl, J., (eds.) Morgan Kaufmann, San Mateo, California, 575–610 (1990).

[6] Wilson, N., A Logic of Partially Satisfied Constraints, *Soft'03: Fifth International Workshop on Soft Constraints*, (attached to *CP'03*), (http://www.sci.unich.it/ bista/organizing/soft-2003/) Kinsale, County Cork, Ireland, 29 September 2003.

[7] Wilson, N., *Soft Constraints with Partially Ordered Preferences*, Unpublished report available on request from the author.