

Composition of Soft Constraints

Kasper Dokter¹, Benjamin Lion¹, and Francesco Santini²

¹ Centrum Wiskunde & Informatica, Amsterdam, Netherlands

² Dipartimento di Matematica e Informatica, University of Perugia, Italy

Abstract. Soft constraint automata extend constraint automata by associating to each action a preference value from a constraint semiring. Such formal and parametric approach can be used to reason according to different metrics of costs. Composition of soft constraint automata

1 Introduction

In this paper, we represent soft constraint automata as logical expressions, called *soft constraint predicates*, that evaluate to a value from a c-semiring. We use this c-semiring value to define a preference.

The compact form of logical expressions makes them very useful for tool like compilers. There is no state/transition space explosion.

We define a generic composition operator on soft constraint predicates over possibly different c-semirings. The main idea is to use a (quotient of the) coproduct in the category of c-semirings. By construction, a coproduct admits embeddings from its constituent c-semirings. This allows us to change the c-semiring of each constituent soft constraint predicate via c-semiring homomorphisms [?].

Soft constraint automata are introduced by Arbab and Santini in [?], where a first formalization and SCA composition has been defined. In the following, different tools based on SCAs have been developed, for instance in [?] a value represents the distance from two SCA, one representing a behavioural-based query and the other SCA the behaviour of a Web service.

Gaducci et al. [?]

We provide an alternative definition of a lexicographic composition of c-semirings as a quotient of the coproduct. The advantage of our new definition over the existing one is that, without any side condition, the constituent c-semirings embed into the new lexicographical product, while this is not always possible for the existing lexicographic composition.

We exploit the fact that such embeddings exist to transform

2 C-semirings

A *c-semiring* [?] is a tuple $(A, +, \times, 0, 1)$ such that

1. A is a carrier set that contains two element $0, 1 \in A$;

2. $+$ is a commutative associative idempotent binary operator, with unit element 0 and absorbing element 1;
3. \times is a commutative associative binary operator that distributes over $+$, with unit element 1, and absorbing element 0.

Well-known instances of c-semirings are the

- *boolean* c-semiring $\mathbb{B} = (\{0, 1\}, \min, \max, 0, 1)$;
- *fuzzy* c-semiring $\mathbb{F} = ([0, 1], \min, \max, 0, 1)$;
- *bottleneck* c-semiring $\mathbb{K} = (\mathbb{R}_{\geq} \cup \{\infty\}, \max, \min, 0, \infty)$;
- *probabilistic* or *Viterbi* c-semiring $\mathbb{V} = ([0, 1], \max, \times, 0, 1)$;
- *weighted* c-semiring $\mathbb{W} = (\mathbb{R}_{\geq} \cup \{\infty\}, \min, +, \infty, 0)$.

Every c-semiring admits a order \leq defined by $r \leq s$ iff $r + s = s$. It is shown in [?] that \leq satisfies the following properties:

1. \leq is a partial order, with minimum 0 and maximum 1;
2. $x + y$ is the least upper bound of x and y ;
3. $x \times y$ is a lower bound of x and y ;
4. (S, \leq) is a complete lattice (i.e., the greatest lower bound exists);
5. $+$ and \times are monotone on \leq .
6. if \times is idempotent, then $+$ distributes over \times , $x \times y$ is the greatest lower bound of x and y , and (S, \leq) is a distributive lattice.

A *morphism* of c-semirings is a map h such that $h(0) = 0$, $h(1) = 1$, $h(x + y) = h(x) + h(y)$, and $h(x \times y) = h(x) \times h(y)$. C-semirings and c-semiring homomorphisms form a category **CS** of c-semirings.

Given two c-semirings $S_i = (A_i, +_i, \times_i, 0_i, 1_i)$, for $i = 0, 1$. Their product $S_0 \times S_1$ is defined by $(A_0 \times A_1, +, \times, (0_0, 0_1), (1_0, 1_1))$, where $(x_0, x_1) + (y_0, y_1) = (x_0 +_0 y_0, x_1 +_1 y_1)$ and $(x_0, x_1) \times (y_0, y_1) = (x_0 \times_0 y_0, x_1 \times_1 y_1)$.

An ideal of a c-semiring $S = (A, +, \times, 0, 1)$ is a subset $I \subseteq A$ such that $0 \in I$, $x + y \in I$ and $a \times x \in I$, for all $x, y \in I$ and $a \in A$. An ideal defines an relation \sim on A via $x \sim y$ if and only if $x + i_x = y + i_y$, for some $i_x, i_y \in I$. It can be shown that \sim is a congruence relation with respect to $+$ and \times , which shows that $+$ and \times define a c-semiring structure on A quotient $S/I = \{A/\sim, \}$

The direct sum $S_0 \oplus S_1$ defined by $S_0 \times S_1$ consists of all linear combinations of elements from S_0 and S_1

Their coproduct $S_0 \cup S_1$ is defined by $\mathbb{B} \oplus S_0 \oplus S_1 \oplus S_0 \otimes_{\mathbb{B}} S_1$ modulo the equations $x + 1 = 1$ and $x \otimes 0 = 0$, for all $x \in E_0 \sqcup E_1$.

2.1 Soft constraint automata

We briefly recall some basic definitions on soft constraint automata.

Definition 1. A *Soft Constraint Automaton (SCA)* is a tuple $(Q, C, \mathbb{E}, \rightarrow, q^0, t)$ such that:

- Q is a set of state, with $q^0 \in Q$;

- C is a set of boolean constraints;
- $x \times y$ is a lower bound of x and y ;
- (S, \leq) is a complete lattice (i.e., the greatest lower bound exists);
- $+$ and \times are monotone on \leq .
- if \times is idempotent, then $+$ distributes over \times , $x \times y$ is the greatest lower bound of x and y , and (S, \leq) is a distributive lattice.

I think that the category of non-trivial ($0 \neq 1$) c-semirings is abelian, with the boolean c-semiring as zero object. Every product of c-semirings is therefore also a coproduct.

3 Soft Constraint Predicates

Defining constraint predicate and c-semiring are usually orthogonal problems. In Soft Constraint Automata, boolean constraints and c-semiring are separately defined and assigned to each transition. Conceptually, we could find some justifications for this separation : the system should first look at the enable transitions (which boolean constraint is true) and chose the best transition (regarding semiring value). Enabling and ordering are two different concerns.

The problem is more practical, and arises during composition. Due to the separation between constraint and c-semiring, the composition operator must differentiate both concerns. In this section, we propose a unified formal model to express both constraint and c-semiring as a soft constraint predicate. Intuitively, a soft constraint predicate is the composition of a c-semiring value from boolean semiring (e.g. the constraint), composed with another c-semiring value (e.g. the semiring value). A new composition operator is lately presented.

3.1 Language of soft constraints

Definition 2. (Language) *The language of soft constraints $\mathcal{L}_{\mathbb{E}}$ is the language of constraints defined as :*

Definition 3. (Predicate) *With L the language of soft constraint, the syntax of a soft constraint predicate (SCP) $\phi_{\mathbb{E}}$ is as follows :*

$$\phi_{\mathbb{E}} := e \mid t_1 = t_2 \mid R(t_1, \dots, t_n) \mid \phi_1 \rightarrow \phi_2 \mid \exists x \phi$$

where $e \in \mathbb{E}$ with \mathbb{E} a c-semiring. Because \mathbb{E} is a lattice, we identify $1_{\mathbb{E}} = \top$ and $0_{\mathbb{E}} = \perp$.

From definition 2, we can derive the definition of other basic logical operators.

- Negation:

$$\neg \phi = \neg \phi \vee \perp = \phi \rightarrow \perp$$

- Conjunction:

$$\phi \wedge \psi = \neg(\neg \phi \vee \neg \psi) = \neg(\phi \rightarrow \neg \psi)$$

Definition 4. (*Modus Ponens*) ϕ and ψ two SCPs. The following holds :

$$\frac{\phi, \quad \phi \rightarrow \psi}{\psi}$$

Definition 5. (*Interpretation of predicates*) For a closed SCP $\phi_{\mathbb{E}} \in \Phi_{\mathbb{E}}$, by induction on $\phi_{\mathbb{E}}$ we define the function $\llbracket \cdot \rrbracket : \Phi_{\mathbb{E}} \rightarrow \mathbb{E}$ that compute the semantic of $\phi_{\mathbb{E}}$ as :

$$\begin{aligned} \llbracket e \rrbracket &= e \in \mathbb{E} \\ \llbracket t_1 = t_2 \rrbracket &= 1_{\mathbb{E}} \quad | \quad 0_{\mathbb{E}} \in \mathbb{E} \\ \llbracket R(t_1, \dots, t_n) \rrbracket &= 1_{\mathbb{E}} \quad | \quad 0_{\mathbb{E}} \in \mathbb{E} \\ \llbracket \phi \rightarrow \psi \rrbracket &= \bigoplus_{\mathbb{E}} \{c \in \mathbb{E} \mid c \otimes_{\mathbb{E}} \llbracket \phi \rrbracket \leq \llbracket \psi \rrbracket\} \end{aligned}$$

where $\bigoplus_{\mathbb{E}}$ is the greatest lower bound operator on \mathbb{E} and $\otimes_{\mathbb{E}}$ the product operator of the c -semiring.

Example with the boolean c -semiring $\mathbb{B} = (\{0, 1\}, \min, \max, 0, 1)$. If we note $\Phi_{\mathbb{B}}$ the set of all formula defined as written in definition 3 :

$$\begin{aligned} \llbracket \top \rightarrow \top \rrbracket &= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} \llbracket \top \rrbracket \leq \llbracket \top \rrbracket\} \\ &= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} 1_{\mathbb{B}} \leq 1_{\mathbb{B}}\} \\ &= 1_{\mathbb{B}} \end{aligned}$$

$$\begin{aligned} \llbracket \top \rightarrow \perp \rrbracket &= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} \llbracket \top \rrbracket \leq \llbracket \perp \rrbracket\} \\ &= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} 1_{\mathbb{B}} \leq 0_{\mathbb{B}}\} \\ &= 0_{\mathbb{B}} \end{aligned}$$

$$\begin{aligned} \llbracket \perp \rightarrow \top \rrbracket &= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} \llbracket \perp \rrbracket \leq \llbracket \top \rrbracket\} \\ &= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} 0_{\mathbb{B}} \leq 1_{\mathbb{B}}\} \\ &= 1_{\mathbb{B}} \end{aligned}$$

$$\begin{aligned} \llbracket \perp \rightarrow \perp \rrbracket &= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} \llbracket \perp \rrbracket \leq \llbracket \perp \rrbracket\} \\ &= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} 0_{\mathbb{B}} \leq 0_{\mathbb{B}}\} \\ &= 1_{\mathbb{B}} \end{aligned}$$

$$\begin{aligned}
\llbracket \neg \psi \rrbracket &= \llbracket \psi \rightarrow \perp \rrbracket \\
&= \bigoplus_{\mathbb{B}} \{c \in \mathbb{B} \mid c \otimes_{\mathbb{B}} \llbracket \psi \rrbracket \leq 0_{\mathbb{B}}\} \\
&= 1_{\mathbb{B}} - \llbracket \psi \rrbracket
\end{aligned}$$

$\top = 0_{\mathbb{W}} \perp = \infty_{\mathbb{W}}$ **Semantics.** The semantic is defined by $\llbracket . \rrbracket$ as follow :

$$\begin{aligned}
\llbracket e \rrbracket &= e \in \mathbb{E} \\
\llbracket t_1 = t_2 \rrbracket &= 1_{\mathbb{B}} \mid 0_{\mathbb{B}} \in \mathbb{B} \\
\llbracket \phi \wedge \psi \rrbracket &= \llbracket \phi \rrbracket \otimes \llbracket \psi \rrbracket \\
\llbracket \phi \vee \psi \rrbracket &= \llbracket \phi \rrbracket \oplus \llbracket \psi \rrbracket \\
\llbracket \neg \phi \rrbracket &= 1_{\mathbb{B}} - \llbracket \phi \rrbracket \quad \text{if } \phi \text{ has a binary semiring valuation}
\end{aligned}$$

3.2 Composition of constraint semirings

Definition 6. The coproduct $E_0 \sqcup E_1$ of two c-semirings E_0 and E_1 is defined by $\mathbb{B} \oplus E_0 \oplus E_1 \oplus E_0 \otimes_{\mathbb{B}} E_1$ modulo the equations $x + 1 = 1$ and $x \otimes 0 = 0$, for all $x \in E_0 \sqcup E_1$.

The main idea is to use quotients of the coproduct as composition operators. The advantage is that by definition we have morphisms from each constituent semiring into their product.

One example of a product is the lexicographical product:

Definition 7. The lexicographical composition of two c-semirings E_0 and E_1 is $E_0 \sqcup E_1$ modulo the equations

$$x_0 \otimes \mathbf{1}_1 + y_0 \otimes \mathbf{0}_1 = y_0 \otimes \mathbf{0}_1,$$

for all $x_0, y_0 \in E_0$ that satisfy $x_0 +_0 y_0 = y_0$.

The distributive law lifts the partial order of each component to the coproduct. Indeed $x + y = y$ implies $x \otimes z + y \otimes z = y \otimes z$. In this way, we obtain an order on the coproduct that relates elements that share at least one component. The rule in Definition 7 extends the order relation on the coproduct by stipulating that the largest element with x_0 in the first component ($x_0 \otimes \mathbf{1}_1$) is smaller than the smallest element with $y_0 \geq x_0$ in the first component ($y_0 \otimes \mathbf{0}_1$).

Our lexicographical product of c-semirings in Definition 7 can be viewed as an generalization of the product defined by Gaducci et al. in [?].

3.3 Composition of soft constraint predicates

Definition 8. The composition of soft constraint predicates ϕ_0/E_0 and ϕ_1/E_1 is $\phi_0 \wedge \phi_1/E$, where E is any coproduct of E_0 and E_1 .

3.4 Restriction

We define a restriction operator on soft constraint automata via *projection* \Downarrow of a constraint satisfaction problem onto a subset of its variables.

Let $c \in \mathcal{C}$ be a constraint and $v \in V$ a variable. The *projection* of c over $V - \{v\}$, written $c \Downarrow_{(V - \{v\})}$, is the constraint c' such that $c'\eta = \sum_{d \in D} c\eta[v := d]$.

4 Related work

Definition of lexicographic product of [?].

Kappé, Arbab and Talcott defined the product of soft constraint automata over composed semirings, such as the lexicographical product [?].

By definition, every c-semiring homomorphism h is order-preserving in the sense that $x \leq y$ implies $h(x) \leq h(y)$. The opposite, $h(x) \leq h(y)$ implies $x \leq y$, is not not always the case, and in this case h is called order-reflecting [?].

In [?] the authors propose an abstraction scheme for soft constraints that uses semiring homomorphism. To find optimal solutions of a SCSP, they first work in the abstract problem and find its optimal solutions, and then use them to solve the concrete problem. In particular, it is shown that a mapping preserves optimal solutions if and only if it is an order-reflecting semiring homomorphism

We refer to [?] for a detailed introduction into semirings.

Semirings are more appropriate than valued structures [?], since valued structures

5 Conclusion

The main results presented in this paper consists in a