Working title

Abstract. Soft constraint automata extend constraint automata by associating, for each transition, a preference value from a csemiring. Preferences model concerns, and multiple concern can be involved in a single choice. We present a new framework to compose preferences of difference csemiring using a co-product. Since constraint automata have an interpretation as a boolean formula, we then investigate the interpretation of soft constraint automata as a semiring formula.

Preferences occur when you have to make a choice. Imagine you are walking on a trekking path, and you meet an intersection. At this point, two actions are possible : turn left, or turn right. Since you don't have any problem with those two actions, you can potentially do both. Suppose, in this case, that no other elements help you to determine your choice, you will chose in a non deterministic way. Now, suppose that you can read the distance from your point to your destination taking the right path or left path. Then, you will be able to chose according to the shortest path. Constraint semiring are suitable structures to define preferences and their composition.

Soft Constraint Automata (SCA) have been developed by Arbab & Santini [1] and constitute a solid ground to design and reason about systems with preferences. Instead of being restricted to crisp constraint, SCA defines soft constraints, based on c-semiring, and can choose the most preferred among a set of choices. Besides a choice operator, composition is also possible and results into new preferences. Defining global preferences by composition of local preferences is a design choice that we do not justify in this paper. Intuitively, the objective is to be able to remove an error of the composed automata by modifying preferences in the composite automata. A diagnosis procedure has been detailed by [3]. Thus, composition is at the core of the definition of Soft Constraint systems.

Until now, several composition operators have been defined (ex: lexicographic, join). The choice of the product space must be done before composition.

In this paper, we first present the definition of a new composition framework for preferences in Soft Constraint Automata, by using the co-product. Composition is decoupled from quotient to product space. Lexicographic or join quotients are defined.

Then, the definition of a new concise model to represent, in a unified way, constraints and preferences into a semiring logic. Logic is extended to any csemiring (not only boolean csemiring).

1 Preferences

We want to get an intuition about how preferences interfere and describe a system's behavior. In the simple example described with west and east actions, the system's behavior can be described by sequences of actions taken by the system. Thus, the sequence $\langle west, east\rangle^\omega$ is one infinite stream allowed by the automaton, representing the case where the robot is alternating between *west* and *east*. Similarly, $\langle stay_{lon}\rangle^\omega$ is also a possible behavior of the system, where the only action taken is $stay_{lon}$. Since constraint automata only tells whether a stream is part of the accepted behavior, we now search for expressing order among accepted behavior. Thus, $\Sigma^\omega \to 2$ becomes $\Sigma^\omega \to A$, where A is a suitable algebraic structure to order the streams.

There exist multiple way to define preferences. An approach used in [**?**] uses multi valued systems. Lattices. Monoids. In [**?**], use of csemiring. The preference in choosing between different mathematical structure for preferences resides in certain properties internal to those structures. We want preferences to be **compositional** and (partially) **ordered**. We will justify in the second section why composition and order are necessary.

Constraint semiring and properties

Semirings constitutes a suitable mathematical structure to define the notion of preferences. A constraint semiring [2] induces an order relation among its element. Therefore, following the initial example on stream and behavior, each action of the stream is attached a preference in a c-semiring. The csemiring will serve as algebraic structure to order streams. We will look at the composition of set of streams with preferences, and the resulting stream order. Given two systems A_1 and A_2 with ordered stream and synchronization constraints (A_1 must synchronize along its run with A_2), what is the ordered stream generated by the parallel composition of A_1 and A_2 ? To answer this question, we must study the properties of the csemiring and order induced by the composition of different csemirings.

A csemiring has an induced order defined given by the properties of its two operators : $+$ and $\times$. The $+$ operator can intuitively be seen as choosing the best value between two value. Thus, if $a, b \in E$, $a + b = a$ can be interpreted as a "is preferred" to b. Alternatively, $\times$ is intuitively used to compose preferences. Then $a \times b$ is a new preference, which could be different from a or b.

Since csemiring define an order among its element, and we have "intuition" behind its operators, we now want to look at the composition of different csemiring, and get intuition behind the order defined on the composed csemiring. Recall that we also want the composition to be ordered, thus being described by a csemiring.

On the search of a structure for our product of csemiring that :

- composed preferences of composite actions.
- induces an order on the composition.
- can later on be composed with a new csemiring.

The carrier of the semiring should contain any possible elements from the composite csemiring. Then, we call the free product such structure, being the cartesian product of all csemiring involved in the composition. It is possible to show that the direct product of two csemiring is itself a csemiring, where $+$ and $\times$ operator are interpreted as $+$ and $\times$ of the underlying csemiring. Thus, if A_1 compose with A_2, where preferences of A_1 are defined over a csemiring $E_1 \times E_2$ and preferences of A_2 are defined over E_3, we get $A = A_1 \times A_2$ with preferences defined over $E_1 \times E_2 \times E_3$.

At this point, remark that the definition of the composition is not without implication. Recall the csemiring induces an order among its element. By defining the product of csemiring as the cartesian product with interpretation of $\times$ and $+$ as underlying csemiring operators, we also define a certain order among the generated streams of the composed system. In this case, the induced order is the following : chose the best preference of the left csemiring, and chose the best preference of the right csemiring, and the resulting preference is the tuple consisting of those two preferences.

Example 1. The case where $\langle west, east\rangle^\omega$ is a possible behavior of the system, and are attached some preferences : $\langle (west, 2), (east, 5)\rangle^\omega$ where 2 and 5 are value of the weighted csemiring. The stream described by $\langle (stay_{lon}, 3)\rangle^\omega$ is also a possible behavior, but since $2 < 3$, the action *west* is preferred to the action $stay_{lon}$, therefore the behavior described by the stream $\langle (west, 2), (east, 5)\rangle^\omega$ is preferred to the behavior described by the stream $\langle (stay_{lon}, 3)\rangle^\omega$. However, taking one step further, in the next element of the stream, we now compare $\langle (east, 5), (west, 2)\rangle^\omega$ with $\langle (stay_{lon}, 3)\rangle^\omega$. Since $3 < 5$, the stream order changed, and the first behavior (which was $\langle (east, 5), (west, 2)\rangle^\omega$) is no more preferred to the other behavior $\langle (stay_{lon}, 3)\rangle^\omega$. The system will now prefer to take the action $stay_{lon}$. Repeating this process, we finally get the stream $\langle (west, 2), (stay_{lon}, 3)\rangle^\omega$ as a preferred behavior.

Example 2. Suppose a composition with a new behavior $\langle (snap, 0.5)\rangle^\omega$. By definition, the composed behavior are $\langle ((west, snap), (2, 0.5)), ((east, snap), (5, 0.5))\rangle^\omega$ and $\langle ((stay_{lon}, \emptyset), (3, \emptyset))\rangle^\omega$

Definition 1. *(semiring) A semiring is a non empty set E on which operations of addition and multiplication have been defined such that:*

- $(E, +)$ *is a commutative monoid with identity element 0.*
- $(E, \times)$ *is a monoid with identity element 1.*
- *Multiplication distributes over addition from either sides.*
- $0 \times e = e \times 0 = 0$ *for all* $e \in E$

We note $\langle E, +, \times, 1, 0\rangle$ to refer to this semiring.

Definition 2. *(csemiring) A csemiring is a semiring $\langle E, +, \times, 1, 0\rangle$ with additional properties :*

- $+$ *is idempotent. We use the notation $\sum(A)$ in prefix notation to describe the sum of all elements of a possibly infinite set $A \subset E$.*
- $\times$ *is commutative.*

- $1 + e = e + 1 = 1$ *for all* $e \in E$

A csemiring admits a partial order $\leq_E$*, defined as the smallest relation satisfying :*

$$\frac{e, e' \in E \quad e + e' = e}{e \leq_E e'}$$

It is shown in [2] that $\leq$ satisfies the following properties:

- $\leq$ is a partial order, with minimum 0 and maximum 1;
- $x + y$ is the least upper bound of x and y;
- $x \times y$ 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.

Intuitively, the $\times$ operator behaves as a composition operator for preferences. The $\sum$ can be seen as the choice of the best preference over a set of preferences, where 1 is the highest preference and 0 is the lowest. We give some well known instances of c-semirings.

Example 3. Examples of c-semirings :

- The Boolean semiring $\mathbb{B}$: $\langle\{\top, \bot\}, \vee, \wedge, 1_{\mathbb{B}}, 0_{\mathbb{B}}\rangle$ where $0_{\mathbb{B}} = \bot$ and $1_{\mathbb{B}} = \top$
- The Weighted semiring $\mathbb{W}$:$\langle \mathbb{N} \cup \{\infty\}, \min, +, 1_{\mathbb{W}}, 0_{\mathbb{W}}\rangle$ where $0_{\mathbb{W}} = \infty$ and $1_{\mathbb{B}} = 0$

Application In the case of our trekker, he first uses boolean semiring to model his choice. If we $left$ and $right$ are propositional variable, we could model the trekker's choice by the following value :

$$TrekChoice = left \vee right$$

Without any information, he can either chose left (make left true) or right.

Now that he has access to the distance, he can chose a weighted semiring instead, and model his choice by the following function :

$$TrekChoice = \begin{cases} left & \text{if } l +_W r = min(l, r) = l \\ right & \text{otherwise} \end{cases}$$

where l is the time it takes on the left path, and r on the right path.

Now that we expressed how to use csemiring to express preferences and choices, we want to be able to model multi criteria choices. Since a csemiring represent one projection of the choice (for instance smallest distance for a trekker), it would also be interesting to express several other dimensions (for instance hardness of the slope, beauty of the landscape, calm of the path, ..). We introduce in the next section the space where product of csemiring are defined : co-product.

Co-product of csemiring

For this section, we assume $E = \langle E, +_E, \times_E, 1, 0\rangle$ and $F = \langle F, +_F, \times_F, 1, 0\rangle$ two c-semirings. We also assume $E \cap F = \mathbb{B}$, where $\mathbb{B} = \langle \{0,1\}, \times, +, 1, 0\rangle$.

Note: 0 and 1 are same objects, shared by all c-semirings. For convenience, we write e_i [resp. f_i] to refer to an element of the csemiring E [resp. F].

Proposition 1. *If $E \cap F \neq \mathbb{B}$, there exists an homomorphism h such that $h(E) \cap F = \mathbb{B}$*

Proof. Define h on E such that h(e) = $\begin{cases} (0, e) & \text{if } e \notin \mathbb{B} \\ e & \text{otherwise} \end{cases}$. The map h is homomorphic to E and $h(E) \cap F = \mathbb{B}$.

Definition 3. *The tensor product $E \otimes_{\mathbb{B}} F$ is the tuple $\langle E \otimes_{\mathbb{B}} F, +, \times, 0, 1\rangle$ where :*

- *$E \subseteq E \otimes_{\mathbb{B}} F$, $F \subseteq E \otimes_{\mathbb{B}} F$ and $E \otimes_{\mathbb{B}} F$ is closed under $+$ and $\times$.*
- *$+$ is idempotent, associative and commutative.*
- *$\times$ is associative and commutative.*
- *$\times$ distributes over $+$.*
- *$\times$ and $+$ are identified on E [resp. F] by $\times_E$ and $+_E$ [resp. $\times_F$ and $+_F$] .*

Lemma 1. *All elements in $g \in E \otimes_{\mathbb{B}} F$ can be written as :*

$$g = \sum_i (e_i \times f_i)$$

where $e_i \in E$ and $f_i \in F$

Proof. Reasoning by induction on the term of $E \otimes_{\mathbb{B}} F$. For all elements $e \in E$ and $f \in F$, since $1 \in E \cap F$, we have:

$$e \times 1 \in E \otimes_{\mathbb{B}} F \quad \text{and} \quad 1 \times f \in E \otimes_{\mathbb{B}} F$$

Suppose $g, h \in E \otimes_{\mathbb{B}} F$, where $g = \sum_i (e_i \times f_i)$ and $h = \sum_j (e_j \times f_j)$ having $e_i, e_j \in E$ and $f_i, f_j \in F$. We look at $g \times h \in E \otimes_{\mathbb{B}} F$ and $g + h \in E \otimes_{\mathbb{B}} F$:

$$g \times h = \sum_i (e_i \times f_i) \times \sum_j (e_j \times f_j) = \sum_{i,j} (e_i \times f_i \times e_j \times f_j) = \sum_{i,j} (e_i \times e_j \times f_i \times f_j)$$

where $e_i \times e_j \in E$ and $f_i \times f_j \in E$,

$$g + h = \sum_i (e_i \times f_i) + \sum_j (e_j \times f_j) = \sum_k (e_k \times f_k)$$

By induction, all terms of the tensor product $E \otimes_{\mathbb{B}} F$ can be represented as a sum over product of elements in E and F.

Theorem 1. *The tensor product $E \otimes_{\mathbb{B}} F$ is the co-product of E and F.*

Proof. We define two injection maps, $\iota_E : E \to E \otimes_{\mathbb{B}} F, e \mapsto e$ and $\iota_F : F \to E \otimes_{\mathbb{B}} F, f \mapsto f$. Given a csemiring G and $h_E : E \to G$ and $h_F : F \to G$ homomorphism, we want to prove the existence and uniqueness of $h : E \otimes_{\mathbb{B}} F \to G$ such that $h_E = h \circ \iota_E$ and $h_F = h \circ \iota_F$.

We define $h : E \otimes_{\mathbb{B}} F \to G, \sum_i (e_i \times f_i) \mapsto \sum_i h_E(e_i) \times h_F(f_i)$. The map h is well defined, since all elements of $E \otimes_{\mathbb{B}} F$ can be written as a sum over product of elements of E and F. Moreover, for all element $e \in E$, $h(\iota_E(e)) = h_E(e) \times h_F(1) = h_E(e)$ and similarly for elements in F. The corresponding diagram commutes. To prove uniqueness of such map, let's define $h' : E \otimes_{\mathbb{B}} F \to G, \sum_i (e_i \times f_i) \mapsto \sum_i h_E(e_i) \times h_F(f_i)$. Then, for all $i \in E \otimes_{\mathbb{B}} F$, $h'(i) = h(i)$, therefore $h = h'$.

Quotients

We are interested in quotients of $E \otimes_{\mathbb{B}} F$ as definition of different products of csemiring.

Definition 4. *(lexicographic) A quotient $h : E \otimes_{\mathbb{B}} F \to L$ is lexicographic if and only if, for all $e_1, e_2 \in E$ and $f_1, f_2 \in F$, there exists $i \in \{1, 0\}$, such that we have:*

$$h(e_1 \times f_1 + e_2 \times f_2) = h(e_i \times f_i)$$

whenever $e_1 + e_2 = e_i$ and $e_1 \neq e_2$; or $f_1 + f_2 = f_i$ and $e_1 = e_2$

Definition 5. *(collapsing elements) We define the set of collapsing element of a semiring E by*

$$\mathcal{C}(E) = \{e \in E \quad | \quad \exists e_1 e_2 \in E, \quad e_1 \times e = e_2 \times e \wedge e_1 \neq e_2\}$$

In other words, it is possible to break a strict inequality after multiplication with a collapsing element. Collapsing element does not preserve strict inequality. Therefore, in the case of some product semiring (for instance lexicographic), distribution of $\times$ over $+$ does not hold. When we come to define product, we should be aware of collapsing element in order to get back a csemiring.

Example : E, F csemirings. $e, e_1, e_2 \in E$ such that $e_1 \times e = e_2 \times e \wedge e_1 < e_2$ and $f_1, f_2 \in F$ with $f_2 < f_1$. We look at a possible element of the coproduct : $e_1 \times f_1 + e_2 \times f_2$. Assuming we have a quotient, and $h : E \sqcup F \to E \times_l F$ can project this term in a lexicographic product space, we would get $h(e_1 \times f_1 + e_2 \times f_2) = (e_2, f_2)$. By distributivity law, and given the homomorphic properties of h, we have :

$$h(e \times (e_1 \times f_1 + e_2 \times f_2)) = h(e \times e_1 \times f_1 + e \times e_2 \times f_2) = h(e \times e_2 \times f_1)$$

and

$$h(e) \times h(e_1 \times f_1 + e_2 \times f_2) = h(e) \times h(e_2 \times f_2) = h(e \times e_2 \times f_2)$$

Thus

$$h(e \times (e_1 \times f_1 + e_2 \times f_2)) \neq h(e) \times h(e_1 \times f_1 + e_2 \times f_2)$$

Lemma 2. *If E is cancellative, then there exists a lexicographic quotient $h : E \otimes_{\mathbb{B}} F \to L$*

2 Semiring Logic

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.

Definition 6. *(Language of soft constraint) A term is defined as:*

$$t := \quad v \quad | \quad f(t_1, ..., t_n) \quad | \quad *$$

A soft constraint over a csemiring $\mathbb{E}$ is a formula ϕ defined by:

$$\phi := \quad e \quad | \quad \phi_1 \times \phi_2 \quad | \quad \phi_1 + \phi_2 \quad | \quad R(t_1, ..., t_n) \quad | \quad \exists x \phi \quad | \quad t_1 = t_2$$

We denote by V_ϕ the set of all free variables contained in a formula ϕ, and by D the data domain of the variables.

Example 4. Examples of c-semirings :

Definition 7. *Csemiring Automaton (CsA) is a tuple $\langle Q, \to, \mathbb{C}, q_0, \rangle$ where :*

- *Q is a set of states and $q_0 \in Q$ is the initial state.*
- *$\mathbb{C}$ is a set of c-semiring formulas representing soft constraints.*
- *$\to \subseteq Q \times \mathbb{C} \times Q$ is a finite relation called the transition relation.*

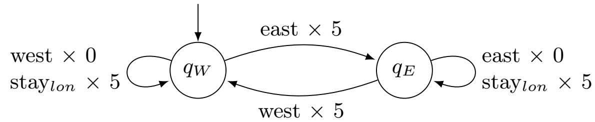

Fig. 1. Constraint automata of a moving agent

Definition 8. *Given two CsA $A_1 = \langle Q_1, \to_1, \mathbb{C}_1, q_{01} \rangle$ and $A_2 = \langle Q_2, \to_2, \mathbb{C}_2, q_{02} \rangle$, their product is $A_1 \times A_2 = \langle Q, \to, \mathbb{C}, (q_{01}, q_{02}) \rangle$ where :*

- *$Q = Q_1 \times Q_2$ and $(q_{01}, q_{02}) \in Q$ is the initial state.*

- *The transition relation* $\rightarrow$ *is the smallest relation satisfying*

$$\frac{q_{01} \xrightarrow{c_1}_1 q'_{01}, \quad q_{02} \xrightarrow{c_2}_2 q'_{02}}{(q_{01}, q_{02}) \xrightarrow{c_1 \times c_2} (q'_{01}, q'_{02})}$$

Example 5. The moving component can be represented as the following formula :

$$\begin{aligned}
\phi = & m = q_1 \times m' = q_0 \times west \times 5 + \\
& m = q_0 \times m' = q_1 \times east \times 5 + \\
& m = q_1 \times m' = q_1 \times east \times 0 + \\
& m = q_0 \times m' = q_0 \times west \times 0 + \\
& m = q_1 \times m' = q_1 \times stay_{lon} \times 5 + \\
& m = q_0 \times m' = q_0 \times stay_{lon} \times 5 \\
\phi = & \sum_i c_i \times w_i
\end{aligned}$$

where w_i is the weight associated to the constraint c_i.

Proposition 2. *The c-semiring automaton* $\langle Q, \rightarrow, \mathbb{C}, q_0 \rangle$*, where* $\mathbb{C}$ *is a set of boolean semiring constraint, is a constraint automaton.*

References

1. Farhad Arbab and Francesco Santini. *Preference and Similarity-Based Behavioral Discovery of Services*, pages 118–133. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
2. Stefano Bistarelli. *Semiring for Soft Constraint Solving and Programming.* Springer, 2004.
3. Tobias Kappé, Farhad Arbab, and Carolyn L. Talcott. A component-oriented framework for autonomous agents. pages 20–38, 2017.