Universiteit Leiden
Opleiding Informatica

Logic for Soft Component Automata

Name:	Tobias Kappé
Date:	15/08/2016
1st supervisor:	Prof.dr.ir. F. Arbab
2nd supervisor:	Dr. M.M. Bonsangue

MASTER'S THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Contents

1 Introduction

Our surroundings are becoming increasingly driven by programs aimed at automating real-world tasks; self-driving cars, crop survey robots and supply chain drones are but modest examples of applications growing continuously more present in our daily lives, in various degrees of visibility. As these *agents* make real-life decisions that have physical consequences for human beings, one looks to formal verification to be able to trust them. Unfortunately, while methods aimed at achieving robustness in uncertain environments appear to be fairly well-established in the context of dynamical systems tasked with steering some continuous signal (see, for example, [34]), the verification of state-based systems seems to lack formalisms that account for fault-tolerance.

What's more, as the inherent complexity of tasks entrusted to autonomous agents grows, so does the description of the agent's behavior, in particular in the context of robustness mechanisms. To keep this complexity manageable, a method based on compositional design is appealing. Such a method should allow a designer to let robustness of the system emerge from the robustness of its individual components as well as the methods used for their composition. Ideally, a compositional system should then be verifiable in a compositional manner: by verifying assertions on components, we should be able to verify assertions on compositions thereof. Moreover, when a system fails to verify a certain property, it is useful to gain some information as to which subset of components were responsible for the behavior that violated said property. To make assertions about the system or its components in particular circumstances, we are also interested in investigating the (optimal) behavior of a system whose actions are constrained to some set of action sequences allowed by the environment.

In this thesis, we propose an automata-based formalism for modeling an autonomous system, which can be seen as a generalization of (Soft) Constraint Automata [4, 1], where actions are endowed with a (possibly multidimensional) value that indicates the preference of the component to perform the action. If the most-preferred action is unavailable, the agent can opt to execute an action of lesser preference. It seems intuitively clear that the presence of alternative (lower-preference) actions makes an agent more robust in coping with situations that do not match its ideal operating conditions (i.e., those compatible with its best-preference actions). We present a set of operators aimed at composing these automata, depending on the concerns they represent.

Verification is approached from two angles. Firstly, we explore an approach based on Linear Temporal Logic (LTL), which builds heavily upon existing methods; here, behavior is constrained based on a minimum preference value. With this LTL-based approach, we can identify components responsible for behavior that violates a desired property. Secondly, we propose a novel method reminiscent of Propositional Dynamic Logic (PDL); here, we are interested in verifying claims about the optimal behavior (in terms of preference) exhibited by the system when constrained to some (sequences of) actions. For the PDL-based approach, we propose two partial orders for deciding optimality, which incorporate the possibility of the system *idling*, and explore model checking for a limited subcase of one such order.

The remainder of this thesis is organized as follows. In Section 2, we list some existing work related to the material on our subject, and contrast it with our work. In Section 3, we discuss the necessary notation. In Section 4 we present our model of actions and their composition; in Section 5 we introduce our automata-based formalism. We then delve into the two complimentary approaches for verification, with the LTL-based verification in Section 6 and the PDL-based verification in Section 7. We present our conclusions in Section 8 and directions for further work in Section 9. To retain focus on the material at hand and not get lost in details, some of the proofs of intermediate lemmas in Section 3 and Section 7 are postponed until the appendices.

1.1 Acknowledgements

The references in this thesis were compiled with the help of the dblp computer science bibliography,[1] as provided by the Computer Science department of the University of Trier and Schloss Dagstuhl.

Over the course of the last year, a number of discussions with Francesco Santini on the finer points of c-semirings and Soft Constraint Satisfaction Problems have proven to be very useful; thanks to him, I was able to develop an intuition of how to use c-semirings to model preferences.

I would also like to thank Marcello Bonsangue (my second supervisor) for the patience he displayed when I tried to summarize my definitions, propositions and conjectures on the whiteboard and scrap paper in his office; his technical acumen has been very inspiring. The members of the Formal Methods group at CWI never failed to make lunch- and coffee-breaks amusing, and provided some very welcome distraction.

Lastly and perhaps most importantly, I would be remiss not to thank Farhad Arbab (my first supervisor) and Carolyn Talcott. Our regular discussions both on Skype and at CWI have been instrumental in steering this work. Carolyn has been very helpful in stressing the importance of running examples and toy models, which gave a firm substrate to discuss our directions. I also benefited enormously from Farhad's optimism and can-do attitude; it was Farhad who always found a positive way to incorporate new lines of reason into the purpose of this work, and this thesis is much better off for it.

[1] http://dblp.uni-trier.de/

2 Related work

The work in this thesis uses the theory of c-semirings as proposed by Bistarelli et al. [6, 9]. Also deserving of mention is [27], which presents an algebraic framework similar to c-semirings called *valuation structures*; these almost match c-semirings, save for the fact that they require a total order. We refer to [7] for a comparison.

To be able to model multiple dimensions of preference, we draw upon [10, 14, 8] to derive techniques for composing c-semirings in a pointwise and lexicographic manner; this approach contrasts [17], which proposes a slightly more complicated structure called a *c-system of monoids*. When composing c-semirings, we need a method to transition smoothly between different c-semirings; to this end the notion of a *homomorphism* is applied to c-semirings; an additional condition on homomorphisms that preserves maximality is found in [21, 15].

It should be noted that in [14], the notion of a *bounded semiring valuation structure* corresponds to our notion of a c-semiring. Also in [14] a generalization of c-semirings called a *partially-ordered valuation structure* is proposed; we make no use of this generalization, as it does not require the existence of a unique least upper bound for each subset, which is something we rely upon to develop material in Section 7.

Preference values used to ensure robustness of autonomous systems can be found in [31]. The use of c-semiring values as labels of transitions can also be found in [1]. It should be noted that, in the latter work, preference values are used as input to preference-based queries to discover similarly-structured automata in a database; in contrast, this thesis uses preference values to drive the behavior of the agent modeled by an automaton. Earlier and slightly less general forms of the work presented in this thesis, particularly those in Section 5, appear in [18].

Our notion of *preference* differs from that of *priority* in process algebra [12, 13] in that priority is used exclusively to ensure that prioritized actions are selected over non-prioritized actions; i.e., a non-prioritized action is never performed in lieu of a prioritized action. In contrast, preferences exist in a possibly broad spectrum of values and may arise compositionally, i.e., from the preferences of constituent actions. Moreover, prioritized actions in process algebra do not compose with non-prioritized actions, whereas composability of our actions is independent of their preference (although some actions may compose more preferably than others).

For the part of this thesis concerned with logic, we make use of Linear Temporal Logic (LTL) [24], in particular the work that relates model checking to automata [33]. An application of LTL to Constraint Automata (CA) appears in the work of Baier et al. [2, 3] as the logic $\mathrm{LTL}_{\mathrm{IO}}$; since our automata formalisms can be considered a generalization of CAs, our use of LTL can likewise be considered as a generalization of $\mathrm{LTL}_{\mathrm{IO}}$. We also draw inspiration from Propositional Dynamic Logic [25, 16] to investigate the optimal behavior of our agents.

The use of c-semirings in combination with logic is not new. In work like [22, 23], however, formulas of a logic are interpreted *over* a c-semiring. This thesis does use c-semirings to establish semantics, but only to compare the preferences of certain behaviors, not to establish a quantitative satisfaction level of a formula.

The term *preference* appears in [32], in the context of Epistemic Modal Logic. Here, the authors are concerned with updating the preferences of an agent based on new knowledge received. While this aim is certainly not unrelated to the applications of the work presented in this thesis, we make the simplifying assumption that the knowledge of our agents is completely described by their state, which in turn completely determines preferences attached to their actions. The act of updating preferences is modeled by the transition into a new state.

3 Preliminaries

In an effort to keep this thesis self-contained, we give a brief overview of the notation and terms used. We start off with some basic mathematical notions in Subsection 3.1 and continue with definitions from the theory of c-semirings in Subsection 3.2. Finally, we review some relevant material on Büchi-automata in Subsection 3.3

3.1 Common mathematical notation

The material in this section will be very familiar and perhaps even uninteresting to the reader with a formal training in Computer Science or Mathematics, as most material discussed here is firmly engrained in canon and can therefore be skipped easily without risking confusion. An exception is formed by the paragraphs on operators and streams, where respectively the notion of *operator* is slightly generalized and some convenient notation for infinite series called *streams* is borrowed from [26].

Inference rules An *(inference) rule* is a compact way of writing down an implication. Such a rule is depicted by writing down the *premises* above a horizontal line, with the *consequence* below said line. As an example of an inference rule, consider the classic syllogism

$$\frac{\text{All humans are mortal} \qquad p \text{ is human}}{p \text{ is mortal}}$$

Assuming the rule above holds, instantiating p, one can apply it in all situations where the premises hold; for example, since all humans are indeed mortal, and Socrates is human, it follows that Socrates is mortal.

Sets A *set* is understood to be a (possibly infinite) collection of objects. In the remainder of this section, we use capital letters X, Y, Z to denote general sets and lower-case symbols x, y, z to denote their elements. The unique set that does not contain any elements is called the *empty set* and denoted by the symbol $\emptyset$. When X is a set and x is an object *contained* in X, we write $x \in X$ and say that x is an *element* of X. Two sets are presumed equal when all elements of either set are also elements of the other.

When X and Y are sets such that when $x \in X$ it holds that $x \in Y$, i.e., all elements of X are also elements of Y, we call X a *subset* of Y and write $X \subseteq Y$. When there also exists *at least one* element of Y that is not an element of X, we say that X is a *strict subset* of Y, written as $X \subset Y$; in this case, X is said to be *smaller* than Y, while Y is *larger* than X. When X and Y are sets, we write $X \cup Y$ for their *union*, i.e., the smallest set containing all elements of X as well as all elements in Y. We furthermore write $X \cap Y$ for their *intersection*, i.e., the largest set containing elements both in X and Y. When the intersection of X and Y is the empty set, we say that X and Y are *disjoint*. We write $X \setminus Y$ for the largest subset of X disjoint with Y. When $x \in X$ holds for a finite number of distinct x, we refer to X as *finite*; if this is not the case, then X is *infinite*.

We can write down a finite set explicitly using curly braces, e.g., $\{x, y\}$ for the set whose elements are x and y exclusively. Note that an element can occur in a set at most once, and the particular order of elements is of no importance; consequently, we identify $\{x, y\}$ with $\{y, x\}$ and $\{x, x\}$ with $\{x\}$. We can define some infinite sets such as the set of natural numbers $\mathbb{N}$ like so: $\{0, 1, 2, 3, \dots\}$; the remaining elements of the set are left implicit. Sets can also be written down in set-builder notation, i.e., $X = \{y \in Y : \phi(y)\}$, where $\phi(y)$ is a description of the elements y of Y that qualify for inclusion in X.

Another convenient method for defining sets, which we employ often in this thesis, is based on inference rules. Specifically, we can define a set as the *unique smallest set* that satisfies one or more inference rules[2], i.e., the unique set that satisfies the inference rules such that none of its subsets does, too. For example, we could have chosen to define the union of X and Y (see above) to be the smallest set $X \cup Y$ satisfying

$$\frac{z \in X \text{ or } z \in Y}{z \in X \cup Y}$$

One immediate advantage of the use of inference rules to define a set X is that if $x \in X$, we immediately know that the premises of *at least* one of the inference rules defining X must hold for x.

We write $X \times Y$ for the *Cartesian product* of X and Y, i.e., the set containing an element $\langle x, y\rangle$ for all elements $x \in X$ and $y \in Y$. Such elements $\langle x, y\rangle$ are called *tuples*. In this thesis, we do not distinguish between $X \times (Y \times Z)$ and $(X \times Y) \times Z$; while, strictly, these are different sets, it should be clear that there exists a unique correspondence between their elements. Accordingly, we drop the parentheses and simply write $X \times Y \times Z$. Also, when the correspondence between a set and its elements is clear from the context, we do not distinguish between $X \times Y$ and $Y \times X$. When we write $\langle x, y\rangle \in X \times Y$, it is implicit that $x \in X$ and $y \in Y$ hold; furthermore, $\langle x, y\rangle = \langle x', y'\rangle$ if and only if $x = x'$ and $y = y'$.

Existential and universal quantifiers For brevity, especially within inference rules or set builder notation, we may replace frequent occurrences of *there exists an $x \in X$, such that* and *for all $x \in X$, it holds that* using symbols $\exists$ and $\forall$ respectively. The assertion $\exists x \in X.\ \forall y \in Y.\ \phi(x, y)$ is thus read as *there exists an element $x \in X$, such that for all elements $y \in Y$, $\phi(x, y)$ holds*, where $\phi(x, y)$ is an assertion dependent on x and y.

Decidability In this thesis, we use the term *decidable* somewhat loosely to indicate any claim that is verifiable algorithmically. For example, the claim $2 + 2 = 4$ is decidable, because we can compute the sum on the left-hand side and verify that it equals the right-hand side.

Relations A *(binary) relation* between sets X and Y is a subset of $X \times Y$. When R is such a subset, we write $x \mathrel{R} y$ as shorthand for $\langle x, y\rangle \in R$; in this case, x and y are said to be *related* by R. When $R \subseteq X \times X$, we say that R is a relation *on* X. When $R \subseteq X \times Y$ and $S \subseteq Y \times Z$ are relations such that $x \mathrel{R} y$ and $y \mathrel{S} z$, then we may *chain* this assertion by writing $x \mathrel{R} y \mathrel{S} z$ instead. Similarly, we write $x \mathrel{R} y, y'$ instead of $x \mathrel{R} y$ and $x \mathrel{R} y'$.

If $R \subseteq X \times Y$ is a relation, we write $\not{R}$ for the *negation* of R, i.e., the unique relation $\not{R} \subseteq X \times Y$ such that for all $x \in X$ and $y \in Y$, we have that $x \mathrel{\not{R}} y$ if and only if $x \mathrel{R} y$ does *not* hold. In a slight abuse of notation, we write $x \notin X$ when x is *not* an element of X.

If R is a relation on X such that for all $x \in X$ it holds that $x \mathrel{R} x$, then R is said to be *reflexive*. If R is a relation on X such that for all $x, x' \in X$ it holds that $x \mathrel{R} x'$ if and only if $x' \mathrel{R} x$, then R is said to be *symmetric*; when $x \mathrel{R} x'$ and $x' \mathrel{R} x$ imply that $x = x'$, R is *antisymmetric*. If R is a relation on X such that for all $x, x', x'' \in X$, if $x \mathrel{R} x'$ and $x' \mathrel{R} x''$, then $x \mathrel{R} x''$, then R is *transitive*. If R is a relation, we may refer to the smallest reflexive (symmetric, transitive) relation containing R as the *reflexive (symmetric, transitive) closure* of R; such a relation always exists, by the same reasoning we use to define sets based on inference rules.

[2] By the Knaster-Tarski theorem, a necessary and sufficient condition for this technique to work (i.e., for a smallest set satisfying the rules to exist uniquely) is that the inference rules should induce a *monotone operator* on sets. The specifics of this are fascinating, but go beyond the scope of this thesis. Instead, we summarize them as follows: *membership of x in the set should never allow using the inference rules to disprove membership of x in the set.* All inference rules used to define sets in this thesis satisfy this condition.

A relation $\trianglelefteq$ on X such that $\trianglelefteq$ is reflexive, antisymmetric and transitive is a *partial order*. When furthermore for all $x, x' \in X$ we have that either $x \trianglelefteq x'$ or $x' \trianglelefteq x$ holds, $\trianglelefteq$ is a *total order*. An example of a total order is the relation $\leq$ on $\mathbb{N}$. As a convention, when we use a symbol like $\trianglelefteq$ to denote a partial order on X, we use the symbol $\triangleleft$ to denote the relation such that $x \triangleleft x'$ if and only if $x \trianglelefteq x'$ and $x \neq x'$. This convention carries over to other symbols used for partial orders, e.g. $\preceq$ versus $\prec$, et cetera.

Functions A relation $R \subseteq X \times Y$ is *functional* when for all $x \in X$ there is precisely one $y \in Y$ such that $x \mathrel{R} y$. We also refer to such a relation as a *function* and use lower-case letters f, g, h to denote functions. If $f \subseteq X \times Y$ is a function, we write $f : X \to Y$ and refer to X as the *domain* of f and Y as the *range* of f.

When $f : X \to Y$ is a function and $x \in X$, we also write $f(x)$ for the unique $y \in Y$ such that $x \mathrel{f} y$ and say that x is *mapped* to y by f. Note that we can uniquely define a function $f : X \to Y$ by specifying which $y \in Y$ is related to each $x \in X$. If $f : X \times Y \to Z$ is a function, we abbreviate by writing $f(x, y)$ instead of $f(\langle x, y\rangle)$. We note that for functions $f, g : X \to Y$, it holds that $f = g$ if and only if $f(x) = g(x)$ for all $x \in X$.

When $f : X \to Y$ and $g : Y \to Z$ are functions, their composition, written $g \circ f$, is the function $g \circ f : X \to Y$ defined by $g(f(x))$ for $x \in X$. When $f : X \to X$ is a function, we define f^0 as the function that maps every element of X to itself, and for $n \in \mathbb{N}$ we define $f^{n+1} = f^n \circ f$. We write Y^X for the set of all functions that have X as domain and Y as range, i.e., $f \in Y^X$ if and only if f is a function $f : X \to Y$. As a special case, if we consider 2 to be the two-element set $\{0, 1\}$, we identify 2^X with the set of all *subsets* of X: if $f \in 2^X$, then f uniquely corresponds to a subset X' of X such that $x \in X'$ if and only if $f(x) = 1$; similarly, we can also obtain a function $f : X \to 2$ from a subset X' of X. Abusing notation, we write 2^X_ω for the set of *finite* subsets of X.

When $f : X \to Y$ is a function and X' is a set, we write $f\restriction_{X'}$ for the *restriction* of X to X', i.e., $f \cap (X' \times Y)$; note that X' is not necessarily a subset of X and that $f\restriction_{X'} : X \cap X' \to Y$ is again a function. When $f : X \to Y$ is a function and $X' \subseteq X$, we write $f(X')$ for the *image* of X' under f, i.e., the set $\{f(x') \in Y : x' \in X'\}$. Similarly, when $Y' \subseteq Y$ we write $f^{-1}(Y')$ for the *inverse image* of Y' under f, i.e., the set $\{x \in X : f(x) \in Y'\}$.

When $X_1 \times X_2$ is a set and $i \in \{1, 2\}$, we write $\mathsf{Pr}_i : X_1 \times X_2$ for the *projection function*, which maps $\langle x_1, x_2\rangle \in X_1 \times X_2$ to x_i. In accordance with earlier notation, we note that $\mathsf{Pr}_i(X_1 \times X_2) = X_i$. Another function that we will use often in the sequel is $\max : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, which takes a pair of natural numbers and returns the largest of the two. Also, when $\trianglelefteq$ is a total order, and S is a finite set, then $\max_{\trianglelefteq}(S)$ is the unique value $s \in S$ such that for all $s' \in S$ it holds that $s' \trianglelefteq s$.

Operators We refer to a function $\circledast : X \times X \to X$ as a *(binary) operator* on X. We commonly write the application of such a function in *infix notation*, i.e., we write $x \circledast y$ instead of $\circledast(x, y)$. When for all $x \in X$ it holds that $x \circledast x = x$, then $\circledast$ is *idempotent*. If for all $x, x' \in X$, it turns out that $x \circledast x' = x' \circledast x$, we call $\circledast$ *commutative*. Similarly, if for all $x, x', x'' \in X$ we know that $x \circledast (x' \circledast x'') = (x \circledast x') \circledast x''$, then $\circledast$ is *associative*.

When $\circledast$ is an associative and commutative operator on X, we immediately obtain a *generalized operator* $\bigcircledast : 2^X_\omega \setminus \{\emptyset\} \to X$, defined for $X' = \{x_1, x_2, \dots, x_n\} \subseteq X$ by $\bigcircledast X' = x_1 \circledast x_2 \circledast \dots \circledast x_n$. Note that the particular order of the x_i does not matter, by commutativity and associativity of $\circledast$. When applying $\bigcircledast$ to some subset of X defined using set-builder notation, we may abbreviate $\bigcircledast\{x : \phi(x)\}$ by writing $\bigcircledast_{\phi(x)} x$.

We occasionally use the enlarged symbol $\bigcircledast$ to denote an operator $2^X \to X$, i.e., defined on possibly infinite subsets of X. An example of such an operator would be $\bigcup : 2^{2^X} \to 2^X$, which takes a set of subsets of X and returns the unique smallest set that contains all such sets as subsets. Note that, whether the domain includes infinite subsets or not, $\circledast$ can be obtained from $\bigcircledast$ by defining $x \circledast x' = \bigcircledast\{x, x'\}$. When we define $\circledast$ thusly, it is immediate that $\circledast$ is idempotent, commutative and associative.

When R is a relation on X and $\circledast : R \to X$ is a function, we refer to $\circledast$ as an operator on X *up to* R and similarly use infix notation. We call such an $\circledast$ *idempotent up to* R if R is reflexive and $x \circledast x = x$ for all $x \in X$; $\circledast$ *commutative up to* R if R is symmetric and $x \mathrel{R} x'$ implies $x \circledast x' = x' \circledast x$. Lastly, $\circledast$ is *associative up to* R if for all $x, x', x'' \in X$ such that $x \mathrel{R} x'$ and $x' \mathrel{R} x''$, then $x \mathrel{R} x''$ if and only if $(x \circledast x') \mathrel{R} x''$, if and only if $x \mathrel{R} (x' \circledast x'')$; moreover, if $x \mathrel{R} x''$, then $(x \circledast x') \circledast x'' = x \circledast (x' \circledast x'')$.

Words and languages Let X be a set and let $n \in \mathbb{N}$. A *word x over X of length n* is a tuple $\langle x_1, x_2, \dots, x_n\rangle$ such that for all $1 \leq i \leq n$ it holds that $x_i \in X$. We juxtapose the elements of x when spelling out a word, i.e., $x = x_1 x_2 \cdots x_n$. We write X^n for the set of words over X of length n. Note that the empty tuple $\langle\rangle$ is a word over X of length 0; this *empty word* is given the symbol ϵ. If $x \in X^n$ and $x' \in X^m$, then $x \cdot x'$ (the *concatenation* of x and x') is the word $x_1 x_2 \cdots x_n x'_1 x'_2 \cdots x'_m \in X^{m+n}$. We can lift the concatenation operator to sets, by defining $X \cdot Y = \{x \cdot y : x \in X, y \in Y\}$. A set of words over X (of arbitrary length) is referred to as a *language over* X; we commonly use the letter L to denote a language. The set of words over X is denoted by X^* (the *Kleene closure* of X) and is alternatively defined as $X^* = \bigcup\{X^n : n \in \mathbb{N}\}$.

The languages that can be constructed starting with finite languages, union, concatenation and Kleene closure are *regular languages*. It is well known that for every regular language R, there exists a *finite automaton* A_R that accepts this language, i.e., $L(A_R) = R$, and that it is decidable whether or not a regular language is a subset of another regular language. The constructions for this are beyond the scope of this thesis; we refer to [28] for an excellent discussion of the relevant theory.

If $x \in X^n \subseteq X^*$, then $|x| = n$ is the *length* of x. Lastly, if $x, x' \in X^*$ such that $|x| \leq |x'|$ and for all $1 \leq i \leq |x|$ it holds that $x_i = x'_i$, then x is called a *prefix* of x'; if moreover $x \neq x'$, then x is a *strict prefix* of x'. The set of prefixes of $x \in X^*$ is denoted by $\mathsf{prefix}(x)$.

Lexicographic order Given a partial order $\trianglelefteq_S$ on a set S, one can define the *induced lexicographic order* $\trianglelefteq_{S^*}$ as the smallest relation on S^* that satisfies the rules

$$\frac{w \in S^*}{\epsilon \trianglelefteq_{S^*} w} \qquad \frac{w, x \in S^* \qquad e, f \in S \qquad e \lhd_S f}{e \cdot w \trianglelefteq_{S^*} f \cdot x} \qquad \frac{w, x \in S^* \qquad e \in S \qquad w \trianglelefteq_S x}{e \cdot w \trianglelefteq_{S^*} e \cdot x}$$

Induction Let X be a set and let R be a relation on X. We call R a *well-founded relation* on X if, for any subset X' of X, there exists an element $x'_0 \in X'$ such that for all $x' \in X' \setminus \{x'_0\}$ it holds that $x' \not R\, x'_0$. Well-founded relations are useful, because they give rise to a proof principle called *induction.* If we want to prove that a property $\phi(x)$ holds for all $x \in X$, it suffices to show that the following inference rule holds:[3]

$$\frac{\forall x' \in X.\ \text{if } x' \mathrel{R} x \text{ and } x \neq x' \text{ then } \phi(x')}{\phi(x)}$$

As a concrete case, we can see that the relation $\leq$ on $\mathbb{N}$ is a well-founded relation. To show that a property $\phi(n)$ holds for all $n \in \mathbb{N}$, we should show that $\phi(0)$ holds, and that if $\phi(n)$ holds, then $\phi(n+1)$ holds, too.

Complete lattices A *complete lattice* is a tuple $\langle L, \leq, \bigvee, \bigwedge \rangle$ such that $\leq$ is a partial order on L and $\bigvee, \bigwedge : 2^L \to L$ are operators such that for all $L' \subseteq L$, $\bigvee L'$ is the *least upper bound* of L', i.e., if $\ell \in L$ such that for all $\ell' \in L'$ it holds that $\ell' \leq \ell$, then it also holds that $\ell' \leq \bigvee L' \leq \ell$. Similarly, $\bigwedge L'$ is the *greatest lower bound* of L', i.e., if $\ell \in L$ such that for all $\ell' \in L'$ it holds that $\ell \leq \ell'$, then $\ell \leq \bigwedge L' \leq \ell'$ holds, also.

Streams We write X^ω for the set $X^{\mathbb{N}}$. An element of X^ω is referred to as a *stream* [26] over X, and commonly denoted using the Greek letters μ, ν, et cetera. We call $\mu(0)$ the *head* of the stream and write μ' for the unique stream defined by $\mu'(n) = \mu(n+1)$ for $n \in \mathbb{N}$, referred to as the *tail* or *derivative* [26]. More generally, the k-th derivative of $\mu \in X^\omega$ is the unique stream $\mu^{(k)}$ such that $\mu^{(k)}(n) = \mu(k+n)$. We identify $X^\omega \times Y^\omega$ with $(X \times Y)^\omega$. Accordingly, if $\langle \mu, \nu \rangle \in X^\omega \times Y^\omega$ we may regard $\langle \mu, \nu \rangle$ as a stream; in particular, this allows us to abbreviate $\langle \mu^{(k)}, \nu^{(k)} \rangle$ by writing $\langle \mu, \nu \rangle^{(k)}$.

3.2 Constraint semirings

We now continue by giving the basic definitions we use from the theory of Constraint Semirings, or c-semirings for short. The name *constraint* here originates from their earlier use as algebraic valuation structures for *Soft Constraint Satisfaction Problems* [9]; we use c-semirings as a convenient structure for reasoning about preferences of actions and their compositions. The definition below diverges slightly from [9, 10] in the $\vee$-operator; this is, however, only a slight generalization; most c-semirings in the literature can still be written in the form below.

Definition 1 ([6, Definition 2.1.2]). *A* Constraint Semiring (c-semiring) *is a tuple* $\langle \mathbb{E}, \bigvee, \otimes, \mathbf{0}, \mathbf{1} \rangle$*, such that* $\mathbb{E}$ *is a set with* $\mathbf{0}, \mathbf{1} \in \mathbb{E}$*,* $\bigvee : 2^{\mathbb{E}} \to \mathbb{E}$ *is an operator,* $\otimes$ *is a commutative and associative operator on* $\mathbb{E}$ *and for all* $e \in \mathbb{E}$*,* $E \subseteq \mathbb{E}$ *and* $\mathcal{E} \subseteq 2^{\mathbb{E}}$*, the following hold:*

- $\bigvee\{e\} = e$, $\bigvee \emptyset = \mathbf{0}$ *and* $\bigvee \mathbb{E} = \mathbf{1}$.
- $\bigvee_{E' \in \mathcal{E}} (\bigvee E') = \bigvee (\bigcup\{E' : E' \in \mathcal{E}\})$ *(the* flattening property*).*
- $\mathbf{0} \otimes e = \mathbf{0}$ *and* $\mathbf{1} \otimes e = e$.
- $\otimes$ *distributes over* $\bigvee$*, i.e.,* $e \otimes \bigvee E = \bigvee\{e \otimes e' : e' \in E\}$.

Every c-semiring $\langle \mathbb{E}, \bigvee, \otimes, \mathbf{0}, \mathbf{1} \rangle$ *induces a relation* $\leq_{\mathbb{E}}$ *defined as the smallest relation that satisfies the rule*

$$\frac{e, e' \in \mathbb{E} \qquad e \vee e' = e'}{e \leq_{\mathbb{E}} e'}$$

Lemma 1 ([6, Theorem 2.1.1]). *Let* $\langle \mathbb{E}, \bigvee, \otimes, \mathbf{0}, \mathbf{1} \rangle$ *be a c-semiring. Then* $\leq_{\mathbb{E}}$ *is a partial order. Moreover, for all* $e \in \mathbb{E}$*, we have that* $\mathbf{0} \leq_{\mathbb{E}} e \leq_{\mathbb{E}} \mathbf{1}$.

Proof. For reflexivity, let $e \in \mathbb{E}$ and consider that $e \vee e = e$ by idempotency of $\vee$, thus $e \leq_{\mathbb{E}} e$. For antisymmetry, let $e, e' \in E$ such that $e \leq_{\mathbb{E}} e' \leq_{\mathbb{E}} e$. Then $e' = e \vee e' = e$ by commutativity of $\vee$. For transitivity, let $e, e', e'' \in E$ such that $e \leq_{\mathbb{E}} e' \leq_{\mathbb{E}} e''$. Then $e \vee e'' = e \vee (e' \vee e'') = (e \vee e') \vee e'' = e' \vee e'' = e''$, thus $e \leq_{\mathbb{E}} e''$.

Let $e \in \mathbb{E}$; to see that $\mathbf{0} \leq_{\mathbb{E}} e$, consider that $\mathbf{0} \vee e = \bigvee \emptyset \vee \bigvee\{e\} = \bigvee(\{e\} \cup \emptyset) = \bigvee\{e\} = e$. Similarly, to see that $e \leq_{\mathbb{E}} \mathbf{1}$, consider that $e \vee \mathbf{1} = \bigvee\{e\} \vee \bigvee \mathbb{E} = \bigvee(\{e\} \cup \mathbb{E}) = \bigvee \mathbb{E} = \mathbf{1}$. □

[3] Again, the details are fascinating, but beyond the scope of this thesis.

Let $\langle \mathbb{E}, \bigvee, \otimes, \mathbf{0}, \mathbf{1} \rangle$ be a c-semiring. We call $\mathbb{E}$ the *carrier* and often use $\mathbb{E}$ as the symbol for the c-semiring. In this case, the corresponding operators and constants are denoted by $\bigvee_{\mathbb{E}}$, $\vee_{\mathbb{E}}$, $\otimes_{\mathbb{E}}$, $\mathbf{0}_{\mathbb{E}}$ and $\mathbf{1}_{\mathbb{E}}$; we drop the subscripts when only one c-semiring appears in the context. The relation $\leq_{\mathbb{E}}$ is called the *induced order* of $\mathbb{E}$, in accordance with Lemma 1. Again, we drop the subscript when no confusion is likely. The operator $\bigvee$ is referred to as the *choice* operator, for reasons that will become clear in a moment. The act of applying $\otimes$ is referred to as *composition.* The constants $\mathbf{0}$ and $\mathbf{1}$ are referred to as the *bottom* and *top* elements of $\mathbb{E}$, respectively. We denote c-semirings using double struck capital letters, such as $\mathbb{E}$, $\mathbb{F}$ et cetera.

Intuitively, a c-semiring $\mathbb{E}$ provides us with a set of *preference values* in the carrier, which we can attach to actions. The operator $\vee$ models *choice* among actions; this is reflected in the relation $\leq_{\mathbb{E}}$: if we have two actions α and β with preferences e_α and e_β such that $e_\alpha <_{\mathbb{E}} e_\beta$ (i.e., $e_\alpha \vee e_\beta = e_\alpha$), then β is said to be *preferred over* α. Note, however, that neither $e_\alpha \leq_{\mathbb{E}} e_\beta$ nor $e_\beta \leq_{\mathbb{E}} e_\alpha$ needs to hold in general (i.e., $\leq_{\mathbb{E}}$ need not be a total order); if this is the case, e_α and e_β are said to be *incomparable.* The operator $\otimes$ models *composition* of preferences: when α and β are composable actions, $e_\alpha \otimes e_\beta$ is the preference attached to their composition. This intuition is be made more explicit in Section 5, where we talk about actions and composition in more detail.

It is often convenient to prohibit actions that are assigned the bottom preference; consequently we refer to such actions as *infeasible actions*, while any other action is said to be *feasible.*

Examples Many different c-semirings exist. Although a sizable number of c-semirings can be said to have the same structure, it is often convenient to use a c-semiring that reflects the context of the actions best. We now consider three examples of c-semirings with meaningful differences with regard to their structure, and mention the contexts in which they could be used.

The simplest useful example of a c-semiring is the *Boolean semiring* $\mathbb{B}$, in which

- The carrier is given by the set $\{\bot, \top\}$.
- The choice operator is given for $B \subseteq \mathbb{B}$ by $\bigvee_{\mathbb{B}} B = \top$ if and only if $\top \in B$.
- The composition operator is given for $b, b' \in \mathbb{B}$ by $b \otimes_{\mathbb{B}} b' = \top$ if and only if $b = \top = b'$.
- $\mathbf{0}_{\mathbb{B}} = \bot$ and $\mathbf{1}_{\mathbb{B}} = \top$.

Because the Boolean semiring allows only two levels of preference, it is primarily of interest as a theoretical device. Using the Boolean semiring, we can capture formalisms unconcerned with preferences by assigning the preference $\bot$ to actions that are not permitted, and $\top$ to actions that are.

For an example of a c-semiring with an infinite carrier, consider the *weighted semiring* $\mathbb{W}$, in which

- The carrier is given by the set $\mathbb{N} \cup \{\infty\}$.
- The choice operator is given for $W \subseteq \mathbb{W}$ by
$$\bigvee_{\mathbb{W}} W = \begin{cases} \infty & W = \{\infty\} \\ \min(W \setminus \{\infty\}) & \text{otherwise} \end{cases}$$
  In which $\min(S)$ is understood to be unique smallest number in $S \subseteq \mathbb{N}$.
- The composition operator is given for $w, w' \in \mathbb{W}$ by
$$w \otimes_{\mathbb{W}} w' = \begin{cases} \infty & \infty \in \{w, w'\} \\ w + w' & \text{otherwise} \end{cases}$$
- $\mathbf{0}_{\mathbb{W}} = \infty$ and $\mathbf{1}_{\mathbb{W}} = 0$.

Intuitively, the weighted semiring models that we can assign *weights* from $\mathbb{N} \cup \{\infty\}$ to actions. Consequently, actions with a lower weight are always preferred over actions with a higher weight — note that this means that $\leq_{\mathbb{W}}$ coincides with the familiar relation $\geq$ on natural numbers. Actions with unbounded weight ∞ are infeasible. The weight of a composed action is given by the sum of the weights of the actions.

Up until this point, we have given only examples of c-semirings whose induced order is a total order. To see that c-semirings whose order is not total have their merit, too, we consider an example inspired by [5]. The UNIX *semiring*[4] $\mathbb{U}$ is the c-semiring where

- The carrier is given by the subsets of $P = \{\mathsf{R}, \mathsf{W}, \mathsf{X}\}$, i.e, $\mathbb{U} = 2^P$.
- The choice operator is given for $U = \{u_1, u_2, \ldots, u_n\} \subseteq \mathbb{U}$ by the finite intersection operator $\bigcap : 2^{\mathbb{U}} \to \mathbb{U}$.
- The composition operator is given by the union, i.e, when $u, u' \in \mathbb{U}$ then $u \otimes_{\mathbb{U}} u' = u \cup u'$.
- $\mathbf{0}_{\mathbb{U}} = P$ and $\mathbf{1}_{\mathbb{U}} = \emptyset$.

[4] The UNIX semiring owes its name to the permission bits *read*, *write* and *execute* that appear in UNIX-like operating systems.

When using the UNIX semiring, a preference attached to an action can be interpreted as the privileges (*read*, *write* or *execute*) necessary for executing the action. By its choice of operators, the UNIX semiring models the *principle of least privilege*: an action α is preferred over another action β if and only if the privileges required for α are a strict subset of those required for β. If α and β were to require the privileges $\{\mathsf{R}, \mathsf{X}\}$ and $\{\mathsf{W}\}$ respectively, they would be incomparable. This stands to reason, because read- and execute-permissions on a file may or may not entail a higher level of privilege than writing permissions.[5] Also, the privileges required for a composed action are given by the union of privileges of the component actions, which seems reasonable to assume.

As an additional example of a c-semiring whose carrier consists of sets, we refer to [10, Section 6.6].

Homomorphisms To move smoothly between c-semirings, we need the notion of *homomorphism.* Intuitively, a homomorphism [21] from c-semirings $\mathbb{E}$ to $\mathbb{F}$ is a function that preserves the algebraic structure of its domain.

Definition 2. *Let $\mathbb{E}$ and $\mathbb{F}$ be c-semirings. A homomorphism from $\mathbb{E}$ to $\mathbb{F}$ is a function $h : \mathbb{E} \to \mathbb{F}$ such that:*

- $h(\mathbf{0}_\mathbb{E}) = \mathbf{0}_\mathbb{F}$ *and* $h(\mathbf{1}_\mathbb{E}) = \mathbf{1}_\mathbb{F}$.
- *For all* $E \subseteq \mathbb{E}$, $h\left(\bigvee_\mathbb{E} E\right) = \bigvee_\mathbb{F} h(E)$.
- *For all* $e, e' \in \mathbb{E}$, $h(e \otimes_\mathbb{E} e') = h(e) \otimes_\mathbb{F} h(e')$.

As an example of a homomorphism, consider that there exists a unique homomorphism $\iota_\mathbb{E}$ from the Boolean semiring $\mathbb{B}$ to any other c-semiring $\mathbb{E}$:[6] simply let $\iota_\mathbb{E}$ map $\bot$ to $\mathbf{0}_\mathbb{E}$ and $\top$ to $\mathbf{1}_\mathbb{E}$; the remainder of the conditions are be validated by the additional requirements in Definition 1. For our purposes, however, we need a similar yet subtly different requirement, which is given by the following definition. We refer to [21] for a more extensive discussion of homomorphisms.

Definition 3. *Let $\mathbb{E}$ and $\mathbb{F}$ be c-semirings, $h : \mathbb{E} \to \mathbb{F}$ a function and $e \in \mathbb{E}$. We call h* e-reflecting *when for all $e' \in \mathbb{E}$, $h(e) \leq_\mathbb{F} h(e')$ if and only if $e \leq_\mathbb{E} e'$. We call h simply* reflecting *if h is e-reflecting for all $e \in \mathbb{E}$.*

As an example of an e-reflecting homomorphism for $e \in \mathbb{E}$ for an arbitrary c-semiring $\mathbb{E}$, consider the *threshold function* $\mathbf{t}_e : \mathbb{E} \to \mathbb{B}$ given by

$$\mathbf{t}_e(e') = \begin{cases} \top & e \leq_\mathbb{E} e' \\ \bot & \text{otherwise} \end{cases}$$

When $\leq_\mathbb{E}$ is a total order, $\otimes$ is idempotent and $e \neq \mathbf{0}$, one can show that $\mathbf{t}_e$ is an e-reflecting homomorphism (Lemma 26). In this case, we refer to $\mathbf{t}_e : \mathbb{E} \to \mathbb{B}$ as the *e-threshold homomorphism* of $\mathbb{E}$, or simply the e-threshold map if it is not guaranteed to be a homomorphism. In general, the e-threshold map is not reflecting; consider for example $\mathbf{t}_7 : \mathbb{W} \to \mathbb{B}$. Here, $\mathbf{t}_7(9) = \bot = \mathbf{t}_7(10)$, thus $\mathbf{t}_7(9) \leq_\mathbb{B} \mathbf{t}_7(10)$, even though $9 \leq_\mathbb{W} 10$ does not hold (recall that $\leq_\mathbb{W}$ coincides with $\geq$). Threshold maps are primarily useful when reducing preference-oriented formalisms to those that do not concern themselves with preferences, as we will see in Section 6.

Cancellative elements In order to properly define composition of c-semirings later on in this section, we need to talk about *cancellative elements* [8] of a c-semiring, a somewhat technical concept closely related to *residuation* [8] and *collapsing elements* [14].[7] A cancellative element e of a c-semiring $\mathbb{E}$ is a preference value for which a simple cancellation rule with regard to $\otimes$ holds.

Definition 4. *Let $\mathbb{E}$ be a c-semiring. An element $e \in \mathbb{E}$ is a* cancellative element *if for all $e', e'' \in \mathbb{E}$, when $e \otimes e' = e \otimes e''$ we know that $e' = e''$. The set of cancellative elements of $\mathbb{E}$ is written $\mathcal{C}(\mathbb{E})$, while the set of non-cancellative elements of $\mathbb{E}$ is written $\overline{\mathcal{C}}(\mathbb{E})$. When $\mathcal{C}(\mathbb{E}) = \mathbb{E} \setminus \{\mathbf{0}\}$, we call $\mathbb{E}$ a* cancellative c-semiring.

Examples of cancellative c-semirings include the Boolean semiring $\mathbb{B}$ and the weighted semiring $\mathbb{W}$. The UNIX-semiring $\mathbb{U}$ is non-cancellative, since $\{\mathsf{R}, \mathsf{W}\} \otimes_\mathbb{U} \{\mathsf{W}\} = \{\mathsf{R}\} \otimes_\mathbb{U} \{\mathsf{W}\}$, even though $\{\mathsf{R}, \mathsf{W}\} \neq \{\mathsf{W}\}$. In general, a c-semiring $\mathbb{E}$ whose composition operator $\otimes$ is idempotent is not cancellative [14], with the exception of $\mathbb{B}$.

It is important to stress, however, that even though the only examples we provide for cancellative c-semirings have an induced order that is total, there exist cancellative c-semirings for which this is not the case; we will see an example of this in the remainder of this section, when we define the *join composition* of c-semirings. The converse is true, as well: there exist totally-ordered c-semirings that are not cancellative (refer to Lemma 28).

Composition When different concerns of varying importance need to be modeled by preference values, it is useful to have the tools to construct a c-semiring that reflects those concerns, using simpler c-semirings as building blocks. For example, suppose that we have actions that carry a weight and a UNIX-permission. The following scenarios seem plausible:

(i) We want to choose an action that is *Pareto-optimal*, in the sense that an action is preferred over another action if it improves on either weight or preference, while keeping the other preference as least as good.

[5] Especially when one considers the finer details of UNIX-permissions, such as the `setuid` bit.

[6] In this sense, $\mathbb{B}$ is the initial object in the category of c-semirings and homomorphisms.

[7] As a matter of fact, collapsing elements are exactly the elements that are non-cancellative; refer to Lemma 27.

(ii) We are primarily concerned with choosing the action with the lowest weight, and secondarily with choosing the action that requires the least privileges.

Conceivably, we should be able to construct a c-semiring for both scenario's using the weighted semiring $\mathbb{W}$ and the UNIX-semiring $\mathbb{U}$. In the following, we define a number of composition operators for c-semirings that allow us to accomplish this.

To model scenario (i), we can use the product of c-semirings [10]; this is again a c-semiring (Lemma 29).

Definition 5. *Let $\mathbb{E}$ and $\mathbb{F}$ be c-semirings. Their* product composition*, written $\mathbb{E} \times \mathbb{F}$, is the c-semiring where*

- *The carrier is given by the Cartesian product of the carriers $\mathbb{E} \times \mathbb{F}$.*
- *The choice operator $\bigvee_{\mathbb{E}\times\mathbb{F}} : 2^{\mathbb{E}\times\mathbb{F}} \to \mathbb{E} \times \mathbb{F}$ is given for $S \subseteq \mathbb{E} \times \mathbb{F}$ by*

$$\bigvee_{\mathbb{E}\times\mathbb{F}} S = \left\langle \bigvee_{\mathbb{E}} \mathsf{Pr}_1(S), \bigvee_{\mathbb{F}} \mathsf{Pr}_2(S) \right\rangle$$

- *The composition operator $\otimes_{\mathbb{E}\times\mathbb{F}}$ is given for $\langle e, f\rangle, \langle e', f'\rangle \in \mathbb{E} \times \mathbb{F}$ by*

$$\langle e, f\rangle \otimes_{\mathbb{E}\times\mathbb{F}} \langle e', f'\rangle = \langle e \otimes_{\mathbb{E}} e', f \otimes_{\mathbb{F}} f'\rangle$$

- $\mathbf{0}_{\mathbb{E}\times\mathbb{F}} = \langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}}\rangle$ *and* $\mathbf{1}_{\mathbb{E}\times\mathbb{F}} = \langle \mathbf{1}_{\mathbb{E}}, \mathbf{1}_{\mathbb{F}}\rangle$.

As an instance of the construction above, consider the c-semiring $\mathbb{W} \times \mathbb{U}$; this semiring faithfully models scenario (i). As an example, consider an action α with preference $e_\alpha = \langle 7, \{\mathsf{R}\}\rangle \in \mathbb{W} \times \mathbb{U}$. This action is preferred over the actions β and γ with preferences $e_\beta = \langle 7, \{\mathsf{R}, \mathsf{W}\}\rangle \in \mathbb{W} \times \mathbb{U}$ and $e_\gamma = \langle 13, \{\mathsf{R}, \mathsf{X}\}\rangle \in \mathbb{W} \times \mathbb{U}$ respectively, since:

$$e_\alpha \vee_{\mathbb{W}\times\mathbb{U}} e_\beta = \langle 7, \{\mathsf{R}\}\rangle \vee_{\mathbb{W}\times\mathbb{U}} \langle 7, \{\mathsf{R}, \mathsf{W}\}\rangle = \langle 7 \vee_{\mathbb{W}} 7, \{\mathsf{R}\} \vee_{\mathbb{U}} \{\mathsf{R}, \mathsf{W}\}\rangle = \langle 7, \{\mathsf{R}\}\rangle = e_\alpha$$
$$e_\alpha \vee_{\mathbb{W}\times\mathbb{U}} e_\gamma = \langle 7, \{\mathsf{R}\}\rangle \vee_{\mathbb{W}\times\mathbb{U}} \langle 13, \{\mathsf{R}, \mathsf{X}\}\rangle = \langle 7 \vee_{\mathbb{W}} 13, \{\mathsf{R}\} \vee_{\mathbb{U}} \{\mathsf{R}, \mathsf{X}\}\rangle = \langle 7, \{\mathsf{R}\}\rangle = e_\alpha$$

Meanwhile, the preference of β is not comparable to γ, since:

$$e_\beta \vee_{\mathbb{W}\times\mathbb{U}} e_\gamma = \langle 7, \{\mathsf{R}, \mathsf{W}\}\rangle \vee_{\mathbb{W}\times\mathbb{U}} \langle 13, \{\mathsf{R}, \mathsf{X}\}\rangle = \langle 7 \vee_{\mathbb{W}} 13, \{\mathsf{R}, \mathsf{W}\} \vee_{\mathbb{U}} \{\mathsf{R}, \mathsf{X}\}\rangle = \langle 7, \{\mathsf{R}\}\rangle \neq e_\beta, e_\gamma$$

Thus, if β and γ were the only available actions, both could be considered *optimal* in terms of preference. Indeed, it is easy to verify that $\leq_{\mathbb{E}\times\mathbb{F}}$ is the *product order* obtained from $\leq_{\mathbb{E}}$ and $\leq_{\mathbb{F}}$ [10].

To cope with scenario (ii), the lexicographic composition of c-semirings [14] can be defined as below. We note that the "most significant" component of a preference in the carrier can be non-cancellative only when the second component is the bottom preference. If this is not done, then the resulting object may fail to be a proper c-semiring, because distributivity of the composition operator over the choice operator fails to hold.

Definition 6. *Let $\mathbb{E}$ and $\mathbb{F}$ be two c-semirings. Their* lexicographic composition*, written $\mathbb{E} \triangleright \mathbb{F}$, is the c-semiring such that*

- *The carrier is the set $(\mathcal{C}(\mathbb{E}) \times \mathbb{F}) \cup (\overline{\mathcal{C}}(\mathbb{E}) \times \{\mathbf{0}_{\mathbb{F}}\})$.*
- *The choice operator is given, for $S \subseteq \mathbb{E} \times \mathbb{F}$, by*

$$\bigvee_{\mathbb{E}\triangleright\mathbb{F}} S = \left\langle \bigvee_{\mathbb{E}} \mathsf{Pr}_1(S), \bigvee_{\mathbb{F}} m(S) \right\rangle$$

in which $m(S)$ contains all and only those elements f of $\mathsf{Pr}_2(S)$ such that $\langle \bigvee_{\mathbb{E}} \mathsf{Pr}_1(S), f\rangle \in S$.

- *The composition operator as well as the bottom and top elements are defined as in Definition 5.*

Intuitively, we can state that the choice operator chooses the value e for the first position between the preferences that occur in the leftmost components of elements of S, after which the second position is chosen amongst $m(S)$, i.e., the preference values that co-occur with e in S. This makes sense, since only the values in S that are not dominated on their first position should contribute towards the second position of $\bigvee_{\mathbb{E}\triangleright\mathbb{F}}$. It is important to note that when $m(S)$ is empty (which may occur when $\leq_{\mathbb{E}}$ is not a total order), $\bigvee_{\mathbb{F}} m(S) = \mathbf{0}_{\mathbb{F}}$.

As a point of order, we note that Definition 6 above appears to contradict [14, Theorem 1] in that $\leq_{\mathbb{E}}$ is not required to be total. However, we believe that such a restriction is not necessary. To the best of our knowledge, a proof of this claim was first provided in [19, Appendix A]; we have reproduced this proof in Theorem 5.

Consider the c-semiring $\mathbb{W} \triangleright \mathbb{U}$; we can use this c-semiring to model scenario (ii). An action α, with preference $e_\alpha = \langle 7, \{\mathsf{R}\}\rangle$, is preferred over the actions β and γ with preferences $\langle 7, \{\mathsf{R}, \mathsf{W}\}\rangle$ and $\langle 10, \emptyset\rangle$ respectively, since

$$e_\alpha \vee_{\mathbb{W}\triangleright\mathbb{U}} e_\beta = \langle 7, \{\mathsf{R}\}\rangle \vee_{\mathbb{W}\triangleright\mathbb{U}} \langle 7, \{\mathsf{R}, \mathsf{W}\}\rangle = \langle 7 \vee_{\mathbb{W}} 10, \{\mathsf{R}\} \vee_{\mathbb{U}} \{\mathsf{R}, \mathsf{W}\}\rangle = \langle 7, \{\mathsf{R}\}\rangle = e_\alpha$$

$$e_\alpha \vee_{\mathbb{W} \rhd \mathbb{U}} e_\gamma = \langle 7, \{\mathsf{R}\} \rangle \vee_{\mathbb{W} \rhd \mathbb{U}} \langle 10, \emptyset \rangle = \left\langle 7 \vee_{\mathbb{W}} 10, \bigvee\nolimits_{\mathbb{U}} \{\{\mathsf{R}\}\} \right\rangle = \langle 7, \{\mathsf{R}\} \rangle = e_\alpha$$

We note that, in the latter case, $m(\{e_\alpha, e_\gamma\}) = \{\{\mathsf{R}\}\}$, because $7 \vee_{\mathbb{W}} 10 = 7$, and $\{\mathsf{R}\}$ is the unique element $u \in \mathbb{U}$ such that $\langle 7, u \rangle \in S$. We thus see that $e_\alpha \leq_{\mathbb{W} \rhd \mathbb{U}} e_\beta, e_\gamma$ matches scenario (ii), since for e_γ the lower weight in the first position takes precedence over the smaller privilege set in the second position, and for e_β the first positions match, but the preferences in the second position of e_α are a strict subset of the preferences in the second position of e_β. Indeed, the order $\leq_{\mathbb{E} \rhd \mathbb{F}}$ can easily be shown to be the *lexicographic order* of $\leq_{\mathbb{E}}$ and $\leq_{\mathbb{F}}$.

One technical objection to the use of the product c-semiring operator of Definition 5 is that it does not preserve cancellativity. Even worse, $\mathbb{E} \times \mathbb{F}$ is *not* cancellative, even if $\mathbb{E}$ and $\mathbb{F}$ are, for the simple reason that

$$\langle \mathbf{0}_{\mathbb{E}}, \mathbf{1}_{\mathbb{F}} \rangle \otimes_{\mathbb{E} \times \mathbb{F}} \langle \mathbf{1}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle = \langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle = \langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle \otimes_{\mathbb{E} \times \mathbb{F}} \langle \mathbf{1}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle$$

As seen in the construction above, non-cancellative elements can occur only in the most significant position of the lexicographic product when the least significant position is the bottom element. Because we need to construct mappings from component c-semirings to composed c-semirings in the sequel (e.g., from $\mathbb{E}$ to $\mathbb{E} \rhd \mathbb{F}$), it is useful to be able to preserve cancellativity of c-semirings when using a product construction. This is done by effectively prohibiting non-cancellative elements to appear in the carrier of the result, as follows.

Definition 7. *Let $\mathbb{E}$ and $\mathbb{F}$ be two cancellative c-semirings. Their* join composition*, written $\mathbb{E} \odot \mathbb{F}$, is the c-semiring where*

- *The carrier is the set $(\mathcal{C}(\mathbb{E}) \times \mathcal{C}(\mathbb{F})) \cup \{\langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle\}$.*
- *The choice and composition operators, as well as the bottom and top elements, are defined as in Definition 5.*

It is now easy to see that $\mathbb{E} \odot \mathbb{F}$ is indeed a c-semiring (Lemma 32) and is cancellative (Lemma 33).[8] Similar to the product of c-semirings, the order of the join $\leq_{\mathbb{E} \odot \mathbb{F}}$ is the product order, but on the restricted carrier. In a sense, $\mathbb{E} \odot \mathbb{F}$ can be regarded as the largest cancellative c-semiring contained in $\mathbb{E} \times \mathbb{F}$; indeed, $\mathcal{C}(\mathbb{E} \times \mathbb{F}) \cup \{\langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle\} = \mathbb{E} \odot \mathbb{F}$.

As hinted in the above, we need a method to embed component c-semirings into a composed c-semiring. To this end, we define the *canonical injections* $\kappa_L^{\mathbb{E} \times \mathbb{F}} : \mathbb{E} \to \mathbb{E} \times \mathbb{F}$ and $\kappa_R^{\mathbb{E} \times \mathbb{F}} : \mathbb{F} \to \mathbb{E} \times \mathbb{F}$ as follows:

$$\kappa_L^{\mathbb{E} \times \mathbb{F}}(e) = \begin{cases} \langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle & e = \mathbf{0}_{\mathbb{E}} \\ \langle e, \mathbf{1}_{\mathbb{E}} \rangle & \text{otherwise} \end{cases} \quad \text{and} \quad \kappa_R^{\mathbb{E} \times \mathbb{F}}(f) = \begin{cases} \langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle & f = \mathbf{0}_{\mathbb{F}} \\ \langle \mathbf{1}_{\mathbb{E}}, f \rangle & \text{otherwise} \end{cases}$$

It can easily be seen that $\kappa_L^{\mathbb{E} \times \mathbb{F}}$ and $\kappa_R^{\mathbb{E} \times \mathbb{F}}$ are reflecting homomorphisms from $\mathbb{E}$ and $\mathbb{F}$ respectively to $\mathbb{E} \times \mathbb{F}$. Moreover, $\kappa_R^{\mathbb{E} \times \mathbb{F}}$ can be seen as a reflecting homomorphism from $\mathbb{F}$ to $\mathbb{E} \rhd \mathbb{F}$, and if $\mathbb{E}$ is cancellative, then $\kappa_L^{\mathbb{E} \times \mathbb{F}}$ can be seen as a reflecting homomorphism from $\mathbb{E}$ to $\mathbb{E} \rhd \mathbb{F}$. Lastly, if $\mathbb{E}$ (respectively $\mathbb{F}$) is cancellative, then $\kappa_L^{\mathbb{E} \times \mathbb{F}}$ (respectively $\kappa_R^{\mathbb{E} \times \mathbb{F}}$) can be seen as a reflecting homomorphism from $\mathbb{E}$ (respectively $\mathbb{F}$) to $\mathbb{E} \odot \mathbb{F}$. We adjust the superscript to denote the range of the canonical injection we are talking about (e.g., $\kappa_L^{\mathbb{E} \rhd \mathbb{F}}$ for the homomorphism from $\mathbb{E}$ to $\mathbb{E} \rhd \mathbb{F}$), but note that their effective mappings remain the same.

Further observations We conclude this section by observing a number of properties of c-semirings to be used later on. The first property we consider is *monotonicity* of the choice and composition operators.

Lemma 2 ([6, Theorem 2.1.2])**.** *Let $\mathbb{E}$ be a c-semiring with e, e', e'' such that $e \leq e'$. Then the choice and composition operators are* monotonic*, i.e., $e \vee e'' \leq e' \vee e''$ and $e \otimes e'' \leq e' \otimes e''$.*

Proof. If $e \leq e'$, then $\bigvee\{e, e'\} = e'$, thus $e \vee e'' \vee e' = \bigvee\{e, e'', e'\} = \bigvee\{e, e'\} \vee e'' = e' \vee e''$, therefore $e \vee e'' \leq e' \vee e'$. For the second claim, we can see that $e' \otimes e'' = (e \vee e') \otimes e'' = (e \otimes e'') \vee (e' \otimes e'')$, thus we know that $e \otimes e'' \leq e' \otimes e''$. □

One immediate application of Lemma 2 is in showing *intensivity*:

Lemma 3 ([6, Theorem 2.1.3])**.** *Let $\mathbb{E}$ be a c-semiring, with $e, e' \in \mathbb{E}$. Then $\otimes$ is* intensive*, i.e., $e' \otimes e \leq e$.*

Proof. By Lemma 1, we know that $e' \leq \mathbf{1}$, thus by Lemma 2 we can conclude that $e \otimes e' \leq e \otimes \mathbf{1} = e$. □

Lemma 3 turns out to be a useful tool in proving a number of properties; we rely on this lemma in Section 7, for example.

In literature concerned with Soft Constraint Satisfaction Problems (SCSPs), such as [6], intensivity means that an additional soft constraint on an SCSP never changes the problem such that a better solution than the best solution to the original problem appears. We exploit intensivity along these lines in Section 6. It should be noted, however, that intensivity does not strictly mean that performing a composition of actions is never preferred over performing a single action; the way we set up our formalisms in Section 5 is such that

[8] We point out that $\mathbb{W} \odot \mathbb{W}$ is an example of a cancellative c-semiring whose induced order is not total.

actions performed by compositions need to be compositions of exactly one action per component, even if some of those actions are effectively non-operations. As such, we trust components to attach a preference value to non-operations in states where other components should be allowed to progress.

Another property that we use in Section 7 is that a c-semiring can be viewed as a complete lattice. More specifically, it holds that the choice operator of a c-semiring can be seen as a lowest upper bound operator, which in turn induces a greatest lower bound operator. The proof of this claim appears in Appendix A.

Lemma 4 ([6, Theorem 2.1.4])**.** *Let $\mathbb{E}$ be a c-semiring. Then there exists a unique operator $\bigwedge : 2^{\mathbb{E}} \to \mathbb{E}$ such that $\langle \mathbb{E}, \leq, \bigvee, \bigwedge \rangle$ is a complete lattice.*

In accordance with Lemma 4, we use the operator $\bigwedge_{\mathbb{E}}$ to denote the greatest lower bound operator of a c-semiring $\mathbb{E}$. As before, we drop the subscript if no confusion is likely.

3.3 Büchi-automata

We now discuss some material from formal language theory used in this thesis. In particular, we use *Büchi-automata* [11], which allow us to give concise descriptions of sets of streams. Moreover, Büchi-automata have a number of pleasant computational and compositional properties, which allow us later on to establish a decision procedure for our logic.

Definition 8. *A* Büchi-automaton[9] *is a tuple $A = \langle Q, \Delta, \to, q^0, F \rangle$ such that*

- *Q is a finite set of* states*, with $q^0 \in Q$ the* initial state *and $F \subseteq Q$ the* accepting states
- *Δ is a finite set of* symbols*, also called the* alphabet *of A*
- *$\to \subseteq Q \times \Delta \times Q$ is a relation called the* transition relation

If $\langle q, \delta, q' \rangle \in \to$, we write $q \xrightarrow{\delta} q'$.

Given a stream $\nu \in \Delta^\omega$ and a Büchi-automaton $A = \langle Q, \Delta, \to, q^0, F \rangle$, we can follow the transitions in A according to the items that appear in ν (provided the transition structure of A admits this), starting in q^0:

$$q^0 \xrightarrow{\nu(0)} q^1 \xrightarrow{\nu(1)} q^2 \xrightarrow{\nu(2)} \cdots$$

In doing so, we obtain a stream of visited states μ, with $\mu(n) = q^n$ for $n \in \mathbb{N}$, called a *trace* (note that there may be more than one such trace obtained from ν). A Büchi-automaton induces a set of streams referred to as its *language*, defined as all streams $\nu \in \Delta^\omega$ that have a trace which visits an accepting state infinitely often.

Definition 9. *Let $A = \langle Q, \Delta, \to, q^0, F \rangle$ be a Büchi-automaton. The* trace relation *of A, written $\rightleftharpoons_A$, and its* infinitary lifting*, written $\rightleftharpoons_A^\omega$, are the smallest relations between Q^ω and Δ^ω satisfying the rules*

$$\frac{\langle \mu, \nu \rangle \in (Q \times \Delta)^\omega \qquad \mu(0) \xrightarrow{\nu(0)} \mu(1)}{\mu \rightleftharpoons_A \nu} \qquad \frac{\forall n \in \mathbb{N}.\ \mu^{(n)} \rightleftharpoons_A \nu^{(n)}}{\mu \rightleftharpoons_A^\omega \nu}$$

The language accepted by A, written $L(A)$, is the smallest set satisfying the rule

$$\frac{\mu \rightleftharpoons_A^\omega \nu \qquad \mu(0) = q^0 \qquad \mu^{-1}(F) \text{ is infinite}}{\nu \in L(A)}$$

Deciding emptiness Interestingly, it is algorithmically decidable whether, given a Büchi-automaton A, $L(A)$ is empty. This comes down to finding an accepting state that can be reached from the initial state with a chain of transitions, and from which a chain of transitions can be taken back to that state. This state can be found using, for example, a depth-first search starting in the initial state for accepting states, followed by a depth-first search starting in every reachable accepting state for a path back to that same state.

Lemma 5 ([33, Proposition 11])**.** *Let $A = \langle Q, \Delta, \to, q^0, F \rangle$ be a Büchi-automaton. Then $L(A) \neq \emptyset$ if and only if there exist $w \in Q^*$ with $w = w_1 w_2 \cdots w_n$ and $x \in \Delta^*$ with $x = x_1 x_2 \cdots x_{n-1}$ such that*

- *$w_1 = q^0$.*
- *There exists a $n' \in \mathbb{N}$ with $n' < n$ and $w_{n'} = w_n \in F$.*
- *$w_1 \xrightarrow{x_1} w_2 \xrightarrow{x_2} \cdots \xrightarrow{x_{n-1}} w_n$.*

[9] Strictly, the definition here is of a *non-deterministic Büchi-automaton.* We do not use deterministic Büchi-automata in this thesis, as they are strictly less powerful [33].

Proof. Suppose $\nu \in L(A)$. Then there exists a $\mu \in Q^\omega$ such that $\mu \rightleftharpoons^{\omega}_{A} \nu$, $\mu(0) = q^0$ and $\mu^{-1}(F)$ is infinite. Because F is finite and $\mathbb{N}$ is infinite, there exists a $q \in F$ and $n, n' \in \mathbb{N}$ such that $q \in F$, $n' < n$ and $\mu(n') = q = \mu(n)$. We then construct $w = \mu(0)\mu(1)\cdots\mu(n) \in Q^*$ and $x = \nu(0)\nu(1)\cdots\nu(n-1)$. We already know that $\mu(0) = q^0$ and $w_{n'} = \mu(n') = \mu(n) = w_n$. From $\mu \rightleftharpoons^{\omega}_{A} \nu$, we also know that $\mu(0) \xrightarrow{\nu(0)} \mu(1) \xrightarrow{\nu(1)} \cdots \xrightarrow{\nu(n-1)} \mu(n)$, or, equivalently, that $w_0 \xrightarrow{x_0} w_1 \xrightarrow{x_1} \cdots \xrightarrow{x_{n-1}} w_n$. The proof in the other direction is similar. □

Note that Lemma 5 provides us with a $\nu \in L(A)$ in case $L(A)$ is non-empty, which can serve as a *counterexample* to the assertion that $L(A)$ is empty. We use this later on in Section 6.

Closure properties Languages accepted by Büchi-automata are also surprisingly robust under set operations, such as intersection, although the constructions involved tend to be somewhat non-trivial. We highlight two useful instances.

Lemma 6 ([33, Proposition 6])**.** *Languages accepted by Büchi-automata are closed under intersection. More precisely, let $A_i = \langle Q_i, \Delta, \to_i, q_i^0, F_i \rangle$ be a Büchi-automaton for $i \in \{0, 1\}$. Then we can construct a Büchi-automaton A such that $L(A) = L(A_0) \cap L(A_1)$.*

Proof. We construct the Büchi-automaton $A = \langle 2 \times Q_0 \times Q_1, \Delta, \to, \langle 0, q_0^0, q_1^0 \rangle, F \rangle$, in which $\langle i, q_0, q_1 \rangle \in F$ if and only if $q_0 \in F_0$ or $q_1 \in F_1$, and the transition relation $\to$ is the smallest relation satisfying the rules

$$\frac{q_0 \xrightarrow{\delta}_0 q_0' \qquad q_1 \xrightarrow{\delta}_1 q_1' \qquad \langle i, q_0, q_1 \rangle \notin F}{\langle i, q_0, q_1 \rangle \xrightarrow{\delta} \langle i, q_0', q_1' \rangle} \qquad \frac{q_0 \xrightarrow{\delta}_0 q_0' \qquad q_1 \xrightarrow{\delta}_1 q_1' \qquad \langle i, q_0, q_1 \rangle \in F}{\langle i, q_0, q_1 \rangle \xrightarrow{\delta} \langle 1-i, q_0', q_1' \rangle}$$

Now, let $\nu \in L(A)$. Then there exists a $\mu \in Q^\omega$ such that $\mu \rightleftharpoons^{\omega}_{A} \nu$. We construct the stream $\mu_i \in Q_i^\omega$ as follows: if $\mu(n) = \langle i, q_0, q_1 \rangle$, then $\mu_i(n) = q_i$. Let $n \in \mathbb{N}$, then $\mu(n) \xrightarrow{\delta} \mu(n+1)$, thus consequently $\mu_i(n) \xrightarrow{\delta}_i \mu(n+1)$, and therefore $\mu_i \rightleftharpoons^{\omega}_{A_i} \nu$. Also, $\mu_i(0) = q_i^0$. To see that $\nu \in L(A_i)$, it remains to be shown that $\mu_i^{-1}(F_i)$ is infinite.

If $\mu(n) \in F$, there must exist a $n' > n$ with $\mu(n') \in F$ (otherwise $\mu^{-1}(F)$ would be finite). Let n' be the smallest such n', and write $\mu(m) = \langle i_m, q_0^m, q_1^m \rangle$ for $m \in \mathbb{N}$, then

$$\langle i_n, q_0^n, q_1^n \rangle \xrightarrow{\nu(n)} \langle i_{n+1}, q_0^{n+1}, q_1^{n+1} \rangle \xrightarrow{\nu(n+1)} \cdots \xrightarrow{\nu(n'-1)} \left\langle i_{n'}, q_0^{n'}, q_1^{n'} \right\rangle$$

Since $\mu(n) = \langle i_n, q_0^n, q_1^n \rangle \in F$, we know that $q_i^n \in F_i$ and $i_{n+1} = 1 - i_n$. Moreover, since $\mu(k) = \langle i_k, q_0^k, q_1^k \rangle \notin F$ for $n < k < n'$, we know that $i_k = i_{k+1}$. From this, it follows that $i_{n'} = 1 - i_n$. This implies that for every $n \in \mathbb{N}$ with $\mu(n) = \langle i, q_0, q_1 \rangle \in F$ there exists a $n' \in \mathbb{N}$ with $n' > n$ and $\mu(n') = \langle 1 - i, q_0', q_1' \rangle \in F$, and therefore that, for $i \in \{0, 1\}$ there exist infinitely many $n \in \mathbb{N}$ such that $\mu(n) = \langle i, q_0, q_1 \rangle$ with $q_i \in F_i$, entailing that for $i \in \{0, 1\}$ there exist infinitely many $n \in \mathbb{N}$ such that $\mu_i(n) \in F_i$; it follows that $\mu_i^{-1}(F_i)$ is infinite, thus $\nu \in L(A_i)$, and therefore $\nu \in L(A_0) \cap L(A_1)$.

For the other direction, let $\nu \in L(A_0) \cap L(A_1)$, then there exists for $i \in \{0, 1\}$ a $\mu_i \in Q_i^\omega$ such that $\mu_i(0) = q_i^0$ and $\mu_i \rightleftharpoons^{\omega}_{A} \nu$. We construct the stream $\mu \in Q^\omega$ according to the rules

$$\frac{}{\mu(0) = \langle 0, q_0^0, q_1^0 \rangle} \qquad \frac{\mu(n) = \langle i, q_0^n, q_1^n \rangle \notin F}{\mu(n+1) = \langle i, \mu_0(n+1), \mu_1(n+1) \rangle} \qquad \frac{\mu(n) = \langle i, q_0^n, q_1^n \rangle \in F}{\mu(n+1) = \langle 1 - i, \mu_0(n+1), \mu_1(n+1) \rangle}$$

We can see that $\mu(0)$ is the initial state of A. To see that $\mu \rightleftharpoons^{\omega}_{A} \nu$, let $n \in \mathbb{N}$. If $\mu(n) = \langle i_n, q_0^k, q_1^k \rangle \notin F$, then since $\mu_i(n) \xrightarrow{\nu(n)}_1 \mu_i(n+1)$ for $i \in \{0, 1\}$, by construction of $\to$ it follows that

$$\mu(n) = \langle i, \mu_0(n), \mu_1(n) \rangle \xrightarrow{\nu(n)} \langle i, \mu_0(n+1), \mu_1(n+1) \rangle = \mu(n+1)$$

Similarly, we can find that $\mu(n) \xrightarrow{\nu(n)} \mu(n+1)$ when $\mu(n) \in F$; therefore $\mu \rightleftharpoons^{\omega}_{A} \nu$. To see that $\mu^{-1}(F)$ is infinite, assume towards a contradiction that $\mu(n) \in F$ for finitely many $n \in \mathbb{N}$. Then it follows that for $i \in \{0, 1\}$ we have $\mu_i(n) \in F_i$ for finitely many $n \in \mathbb{N}$, which contradicts that $\mu_i^{-1}(F_i)$ is infinite. We thus conclude that $\mu^{-1}(F)$ is infinite and therefore that $\nu \in L(A)$. □

Note that the construction in Lemma 6 gives us an algorithm to decide whether two Büchi-automata share a common stream, and if so, provides us with an example of such a stream: simply construct the automaton that accepts the intersection of the languages and use Lemma 5 to find out whether this intersection is empty — if not, we obtain a stream accepted by both.

Another closure property enjoyed by Büchi-automata is *complementation*, i.e., given a Büchi-automaton A we can construct a Büchi-automaton that accepts precisely the streams that are *not* accepted by A. The proof for this is somewhat involved and omitted from this thesis; we refer to [30] for an example.

Lemma 7 ([30, Theorem 2.6])**.** *Languages accepted by Büchi-automata are closed under complement: if $A = \langle Q, \Delta, \to, q^0, F \rangle$ is a Büchi-automaton, then we can construct a Büchi-automaton A_C such that $\nu \in L(A)$ if and only if $\nu \notin L(A_C)$.*

We conclude this section by noting a closure property of Büchi-automata that will be useful in Section 6.

Lemma 8. *Let* $A_i = \langle Q_i, \Delta, \rightarrow_i, q_i^0, F_i \rangle$ *be a Büchi-automaton for* $i \in \{0, 1\}$. *Then there exists a Büchi-automaton* A *such that* $\nu \in L(A)$ *if and only if there exists an* n *such that for all* $0 \leq k < n$ *it holds that* $\nu^{(k)} \in L(A_0)$ *and* $\nu^{(n)} \in L(A_1)$.

The proof of Lemma 8 is omitted, for it is beyond the scope of this thesis. To get a general idea, one could prove Lemma 8 by constructing an *alternating Büchi-automaton* [33] accepting the described language. From this alternating Büchi-automaton one can then construct a Büchi-automaton that accepts exactly the same language. We refer to [33] for the translation of alternating Büchi-automata to Büchi-automata.

4 Component Action Systems

To define our formalism for agents, we first need to define precisely what we mean by *actions* and *compositions* thereof. This is captured by the definition below.

Definition 10. *A* Component Action System (CAS) *is a tuple* $\langle \Sigma, \bigcirc, \Box \rangle$ *such that*

- Σ *is a* set of actions.
- *The* composability relation $\bigcirc$ *is a reflexive and symmetric relation on* Σ.
- *The* composition operator $\Box : \bigcirc \to \Sigma$ *is idempotent, commutative and associative on* Σ *up to* $\bigcirc$.

Given a CAS $\langle \Sigma, \bigcirc, \Box \rangle$, we refer to Σ as the *carrier* of the CAS; we use Σ as the symbol for an abstract CAS. If Σ is a CAS, the corresponding relation and operator are denoted by $\bigcirc_\Sigma$ and $\Box_\Sigma$. As with c-semirings, we drop the subscript when only one CAS appears in the context.

We can justify Definition 10 on an intuitive level. Suppose one component of our agent wants to perform the action σ, while another component wants to perform the action τ. If $\sigma = \tau$, the components agree on the action and thus that same action should be performed (hence reflexivity of $\bigcirc$ and idempotency of $\Box$). If $\sigma \neq \tau$, the actions might still make sense to be executed in composition; they could be wholly unrelated (e.g., σ could be *pick up payload* and τ could be *send heartbeat signal*) or complementary (e.g. σ could be *pick up payload* and τ could be *release payload*). If this is the case, we need a method to derive the composed operation; in the former case, one can probably compose the actions concurrently (e.g. *pick up payload while sending heartbeat signal*), while in the latter case the composed action may constitute a logical consequence of the component actions (e.g. *exchange payloads*).

We require commutativity of the composition operator so as not to distinguish composing action σ with action τ from composing the actions in reverse order. This seems reasonable, as there is no order (temporal or otherwise) between the actions under composition that should make their composition non-commutative — such an ordering would have to arise from an ordering in the components, which is absent from our model.

The condition that the composition operator is associative *up to composability* guarantees that, when composing a number of mutually composable actions, the resulting action does not depend on the particular order of composition. However, it does not preclude the possibility that for actions $\sigma, \tau, \rho \in \Sigma$, action σ is composable with both τ and ρ, but $\sigma \Box \tau$ is not composable with ρ (and $\sigma \Box \rho$ is not composable with τ). Such a situation may occur precisely when $\tau \not\bigcirc \rho$. This, too, seems reasonable: the action *pick up payload* may be composable with both *release payload* and *burn payload*, but *exchange payloads* should be incomposable with *burn payload*, because the latter is incomposable with the component action *release payload*.

A simple CAS We now consider a simple and very concrete CAS called the *movement CAS* $\mathcal{M}$. Suppose we want to model movements on a sphere with a north and south pole. Our set of actions $\mathcal{M}$ could consist of:

- movement generally in the cardinal directions: north, east, south and west
- movement purely in the cardinal directions: $\mathsf{north}^\star$, $\mathsf{east}^\star$, $\mathsf{south}^\star$ and $\mathsf{west}^\star$
- movement in the intercardinal directions: $\mathsf{northeast}$, $\mathsf{northwest}$, $\mathsf{southeast}$ and $\mathsf{southwest}$
- staying on the same latitude or longitude, or both: $\mathsf{stay}_{\mathsf{lat}}$, $\mathsf{stay}_{\mathsf{lon}}$ and stay

It seems reasonable that movement that does not make assertions about longitude should be composable with longitudinal movement; for example, the action east should be composable with the actions north and $\mathsf{stay}_{\mathsf{lat}}$. Also, movement purely in one cardinal direction should be composable with actions that require movement in that same cardinal direction. Lastly, actions are inherently composable with themselves. All of these requirements are modeled by defining the composability relation $\obar$ to be the reflexive and symmetric closure of the smallest relation satisfying the respective rules

$$\frac{m \in \{\mathsf{north}, \mathsf{south}, \mathsf{stay}_{\mathsf{lat}}\} \qquad m' \in \{\mathsf{east}, \mathsf{west}, \mathsf{stay}_{\mathsf{lon}}\}}{m \obar m'} \qquad \frac{m \in \{\mathsf{north}, \mathsf{south}, \mathsf{east}, \mathsf{west}\}}{m \obar m^\star}$$

The composition operator $\boxbar$ can now be defined to match our intuition; let $m, m' \in \mathcal{M}$, then:

- If $m = m'$, then $m \boxbar m' = m$.
- If $m \in \{\mathsf{north}, \mathsf{south}\}$ and $m' \in \{\mathsf{east}, \mathsf{west}\}$, then $m \boxbar m' = m' \boxbar m$ is the corresponding intercardinal direction (e.g., $\mathsf{northeast}$ when $m = \mathsf{north}$ and $m' = \mathsf{east}$).
- If $m \in \{\mathsf{north}, \mathsf{south}\}$ and $m' = \mathsf{stay}_{\mathsf{lat}}$ then $m \boxbar m' = m' \boxbar m = m^\star$.
- If $m \in \{\mathsf{east}, \mathsf{west}\}$ and $m' = \mathsf{stay}_{\mathsf{lon}}$ then $m \boxbar m' = m' \boxbar m = m^\star$.
- If $m = \mathsf{stay}_{\mathsf{lon}}$ and $m' = \mathsf{stay}_{\mathsf{lat}}$, then $m \boxbar m' = m' \boxbar m = \mathsf{stay}$.

The proof that $\langle \mathcal{M}, \obar, \boxbar \rangle$ is a CAS is left as an exercise to the reader.

A general CAS While specifying a CAS explicitly gives the designer a lot of control, we can see that even for a (conceptually) simple CAS such as the movement CAS above some care needs to be taken to keep the structure from violating the requirements of a CAS. Most of the time, it is easier to impose a certain structure on the actions and let the composability relation and composition operator arise from this structure. We now present such a CAS, based on the action model of *Constraint Automata* [4, 1], where an action consists of a set of *ports* that *fire*, along with the data values that *flow* at those ports.[10]

In the most general sense, we can imagine that the ports represent sensors and actuators of the agent, and that the datum that flows at a firing port represents a sensor reading or the particular mode of actuation; for example, a port rotate could be connected to a rotor, with the datum ↻ (respectively ↺) representing clockwise (respectively counterclockwise) actuation. On the level of components, we may suppose that a port represents a (possibly shared) channel on which a component may interact with other components, and the datum at the port represents the information that flows through the channel the instant the action is performed. For an extensive case study that uses this paradigm to represent the behavior of an autonomous agent, we refer to [18].

Fix a set of ports $\mathcal{P}$ and a data domain $\mathcal{D}$. For our actions, we choose $\mathcal{A} = \mathcal{D}^{\mathcal{P}}$, i.e., $\mathcal{A}$ consists of the functions from $\mathcal{P}$ to $\mathcal{D}$; every action thus represents an assignment of data to ports. We isolate a special datum $* \in \mathcal{D}$; when $\alpha \in \mathcal{A}$ and $\alpha(p) = *$ for some $p \in \mathcal{P}$, we say that α *does not fire* p; contrarily, when $\alpha(p) \neq *$, α *does fire* p. When $\alpha \in \mathcal{A}$, we write $\mathsf{fire}(\alpha)$ for *firing set* of α, i.e., $\mathsf{fire}(\alpha) = \{p \in \mathcal{P} : \alpha(p) \neq *\}$.

Recall that in Constraint Automata, actions are composable if and only if they agree on common ports [4]. This means that actions $\alpha, \beta \in \mathcal{A}$ are composable when ports that they both fire (i.e., the elements $p \in \mathsf{fire}(\alpha) \cap \mathsf{fire}(\beta)$) carry the same datum (i.e., $\alpha(p) = \beta(p)$). As a result, we can choose our composability relation $\Diamond$ to be the smallest relation satisfying the following rule:

$$\frac{\alpha, \beta \in \mathcal{A} \qquad \alpha\restriction_{\mathsf{fire}(\beta)} = \beta\restriction_{\mathsf{fire}(\alpha)}}{\alpha \mathrel{\Diamond} \beta}$$

Lastly, we need to define the composition operator. Recall that for Constraint Automata, the composition of two composable actions is defined to fire the union of their ports, with the data flow at common ports given by either action, and the data flow at a firing port unique to that action given by the data flow at that port in that action. Formally, this means that our composition operator $\boxtimes$ can be defined for $\alpha \mathrel{\Diamond} \beta$ as follows:

$$(\alpha \boxtimes \beta)(p) = \begin{cases} \alpha(p) & p \in \mathsf{fire}(\alpha) \\ \beta(p) & p \in \mathsf{fire}(\beta) \\ * & \text{otherwise} \end{cases}$$

We conclude this section by showing that the action model used by Constraint Automata is indeed a CAS.

Lemma 9. *$\langle \mathcal{A}, \Diamond, \boxtimes \rangle$, as defined above, is a CAS.*

Proof. It is easy to see that $\Diamond$ is symmetric and reflexive and that $\boxtimes$ is idempotent by inspecting their definitions. For the following, let $\alpha, \beta, \gamma \in \mathcal{A}$. Before we proceed, we observe that from the definition of $\boxtimes$ it follows that if $\alpha \mathrel{\Diamond} \beta$, then $\mathsf{fire}(\alpha \boxtimes \beta) = \mathsf{fire}(\alpha) \cup \mathsf{fire}(\beta)$. To see that $\boxtimes$ is commutative up to $\Diamond$, let $\alpha \mathrel{\Diamond} \beta$ and $p \in \mathcal{P}$. If $p \in \mathsf{fire}(\alpha)$ and $p \in \mathsf{fire}(\beta)$, then $(\alpha \boxtimes \beta)(p) = \alpha(p)$ and $(\beta \boxtimes \alpha)(p) = \beta(p)$. But $\alpha(p) = \beta(p)$ since $\alpha \mathrel{\Diamond} \beta$. Thus $(\alpha \boxtimes \beta)(p) = (\beta \boxtimes \alpha)(p)$. If $p \in \mathsf{fire}(\alpha)$ and $p \notin \mathsf{fire}(\beta)$, then $(\alpha \boxtimes \beta)(p) = \alpha(p) = (\beta \boxtimes \alpha)(p)$; similarly, we find that $\alpha \boxtimes \beta$ and $\beta \boxtimes \alpha$ agree when $p \notin \mathsf{fire}(\alpha)$ and $p \in \mathsf{fire}(\beta)$. The case where $p \notin \mathsf{fire}(\alpha)$ and $p \notin \mathsf{fire}(\beta)$ remains, but then $(\alpha \boxtimes \beta)(p) = * = (\beta \boxtimes \alpha)(p)$. We conclude that $\alpha \boxtimes \beta = \beta \boxtimes \alpha$ and thus that $\boxtimes$ is commutative.

We must also show that $\boxtimes$ is associative up to $\Diamond$, i.e., that when $\alpha \mathrel{\Diamond} \beta$ and $\beta \mathrel{\Diamond} \gamma$, then it follows that $\gamma \mathrel{\Diamond} \alpha$ if and only if $(\alpha \boxtimes \beta) \mathrel{\Diamond} \gamma$, which in turn holds if and only if $\alpha \mathrel{\Diamond} (\beta \boxtimes \gamma)$. Assume that $\alpha \mathrel{\Diamond} \gamma$ and let $p \in \mathcal{P}$. If $p \in \mathsf{fire}(\alpha \boxtimes \beta) \cap \mathsf{fire}(\gamma)$, then either $p \in \mathsf{fire}(\alpha)$ or $p \in \mathsf{fire}(\beta)$. If $p \in \mathsf{fire}(\alpha)$, then $(\alpha \boxtimes \beta)(p) = \alpha(p)$ by definition of $\boxtimes$ and $\alpha(p) = \gamma(p)$ since $\alpha \mathrel{\Diamond} \gamma$, thus $(\alpha \boxtimes \beta)(p) = \gamma(p)$. If on the other hand $p \in \mathsf{fire}(\beta)$, we can derive that $(\alpha \boxtimes \beta)(p) = \gamma(p)$ by a similar reasoning. We conclude that $(\alpha \boxtimes \beta) \mathrel{\Diamond} \gamma$. If on the other hand we assume that $\alpha \mathrel{\Diamond} (\beta \boxtimes \gamma)$, let $p \in \mathsf{fire}(\alpha) \cap \mathsf{fire}(\gamma)$. Then $p \in \mathsf{fire}(\alpha) \cap (\mathsf{fire}(\beta) \cup \mathsf{fire}(\gamma)) = \mathsf{fire}(\alpha) \cap \mathsf{fire}(\beta \boxtimes \gamma)$ immediately, and thus $\alpha(p) = (\beta \boxtimes \gamma)(p) = \gamma(p)$; consequently, $\alpha \mathrel{\Diamond} \gamma$. We have thus established that $\alpha \mathrel{\Diamond} \gamma$ if and only if $\alpha \mathrel{\Diamond} (\beta \boxtimes \gamma)$; we can prove that $\alpha \mathrel{\Diamond} \gamma$ if and only if $(\alpha \boxtimes \beta) \mathrel{\Diamond} \gamma$ in an analogous fashion.

Lastly, we need to show that if α, β and γ are all related by $\Diamond$, then $(\alpha \boxtimes \beta) \boxtimes \gamma = \alpha \boxtimes (\beta \boxtimes \gamma)$. First, observe that these expressions are well-defined, because $(\alpha \boxtimes \beta) \mathrel{\Diamond} \gamma$ and $\alpha \mathrel{\Diamond} (\beta \boxtimes \gamma)$ are true, by the reasoning above. If $p \in \mathsf{fire}(\alpha \boxtimes \beta)$, then $p \in \mathsf{fire}(\alpha)$ or $p \in \mathsf{fire}(\beta)$. In the former case, $((\alpha \boxtimes \beta) \boxtimes \gamma)(p) = (\alpha \boxtimes \beta)(p) = \alpha(p) = (\alpha \boxtimes (\beta \boxtimes \gamma))(p)$ by definition of $\boxtimes$. In the latter case, $p \in \mathsf{fire}(\beta \boxtimes \gamma)$ and thus $(\alpha \boxtimes (\beta \boxtimes \gamma))(p) = (\beta \boxtimes \gamma)(p) = \beta(p) = (\alpha \boxtimes \beta)(p) = ((\alpha \boxtimes \beta) \boxtimes \gamma)(p)$, also by definition of $\boxtimes$. If $p \notin \mathsf{fire}(\alpha \boxtimes \beta)$, then $((\alpha \boxtimes \beta) \boxtimes \gamma)(p) = \gamma(p) = ((\alpha \boxtimes \beta) \boxtimes \gamma)(p)$, regardless of whether $p \in \mathsf{fire}(\gamma)$. □

[10] This is a slight misrepresentation; formally, transitions in Constraint Automata are labelled with a *firing set* of ports and a *data constraint* that represents a constraint satisfaction problem P on the firing set. The data that flows at the ports are required to constitute a solution to P. We argue that this distinction does not matter when we limit ourselves to finite data domains, as any Constraint Automaton with a finite data domain can be represented to match the description above by splitting P into a set of Constraint Satisfaction Problems such that each problem matches exactly one solution to P.

5 Soft Component Automata

We now turn our attention to the formalism used to model components. We propose a framework that can be seen as a generalization of (Soft) Constraint Automata [4, 1], called *Soft Component Automata* in which automata can be composed, and the preferences of their composed actions are the composed preferences of their actions. The term *soft* has been used to describe generalizations that involve adding preferences (c.f. *Soft Constraint Satisfaction* versus *Constraint Satisfaction* [9, 10] and *Soft Constraint Automata* versus *Constraint Automata* [1]). The term *component* emphasizes that these automata are compositional, i.e., we can construct more complex automata from a basic set of simple automata.

Definition 11. *A* Soft Component Automaton (SCA) *is a tuple* $\langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ *such that*

- Q *is a finite set of states with* $q^0 \in Q$ *the* initial state*.*
- Σ *is a Component Action System called the* underlying CAS*.*
- $\mathbb{E}$ *is a c-semiring called the* underlying c-semiring *with* $t \in \mathbb{E}$ *called the* threshold value*.*
- $\rightarrow \subseteq Q \times \Sigma \times \mathbb{E} \times Q$ *is a finite relation called the* transition relation*.*

When $\langle q, \sigma, e, q'\rangle \in \rightarrow$ *we write* $q \xrightarrow{\sigma,\, e} q'$*.*

We interpret the threshold value in one of two ways; in Section 6 it serves as a minimum value for feasible actions, whereas in Section 7, we use it as a value that determines the preference of the agent to perform a special non-operation called *idling*. We ignore the threshold value for the remainder of this section.

We can depict an SCA graphically as a labelled transition system: we draw a graph with a vertex for every state, and edges labeled with elements from $\mathbb{E} \times \Sigma$ between states to represent transitions. Consider, for example, a component responsible for letting an agent patrolling on a sphere, along a path parallel to the equator. We could model this component using the movement CAS $\mathcal{M}$ and the weighted semiring $\mathbb{W}$, as in Figure 1.[11]

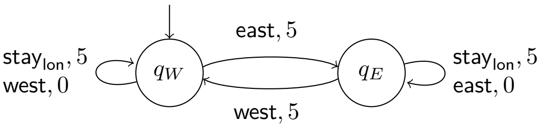


Figure 1: A component modeling a patrolling movement.

The component depicted in Figure 1 starts in state q_W, where it is moving towards the westmost waypoint. As long as it can move west (i.e., as long as the waypoint has not been reached), it will do so, since the action west has the lowest weight and therefore the highest preference. When this action is not available, the component tries to stay on the same longitude, by performing the action stay$_{\text{lon}}$ (hoping that it will be able to progress at some point); it may also try to turn by performing the action east. The state q_E is the dual to q_W: here, the component attempts to move towards the eastmost waypoint.

5.1 Composition

The composition operator we propose for SCAs is defined below. Note that only transitions whose actions are composable give rise to transitions with composed actions; the composed action of such a transition is obtained from the CAS, while the composed preference is provided by the c-semiring.

Definition 12. *Let* $A_i = \langle Q_i, \Sigma, \mathbb{E}, \rightarrow_i, q_i^0, t_i\rangle$ *be an SCA for* $i \in \{0, 1\}$*. The composition of* A_0 *and* A_1*, written* $A_0 \bowtie A_1$*, is the SCA* $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ *such that*

- *The set of states* Q *is* $Q_0 \times Q_1$*, with the initial state* $q^0 = \langle q_0^0, q_1^0\rangle$*.*
- *The threshold value* t *is* $t_0 \otimes t_1 \in \mathbb{E}$*.*
- *The transition relation* $\rightarrow$ *is the smallest relation that satisfies the rule*

$$\frac{q_0 \xrightarrow{\sigma_1,\, e_0}_0 q_0' \qquad q_1 \xrightarrow{\sigma_1,\, e_1}_2 q_1' \qquad \sigma_0 \bigcirc \sigma_1}{\langle q_0, q_1\rangle \xrightarrow{\sigma_0 \,\square\, \sigma_1,\, e_0 \otimes e_1} \langle q_0', q_1'\rangle}$$

By unwinding the definitions of the action system $\langle \mathcal{A}, \Diamond, \boxtimes\rangle$ in combination with Definition 12, one can see that Soft Component Automata properly generalize (Soft) Constraint Automata: transitions compose if and only if they agree on common ports, and the composed action consists of all ports involved firing in concert.

[11] When depicting SCAs graphically, the CAS and c-semiring used should be obvious from or specified in the context.

Let us first consider an example of the composition operator. Let A_{move} be the SCA in Figure 1. Suppose our agent can also diverge from the path (by moving north or south) but we prefer the agent to stay on track. A possible component tasked with this responsibility, A_{diverge}, is depicted in Figure 2; this SCA, too, uses the movement CAS and the weighted semiring. The initial state of this component q_M signifies that the agent is centered on the path. As long as the agent can remain centered, it will do so by executing the action $\mathsf{stay}_{\mathsf{lat}}$; if this is not possible, it tries to diverge from the path to either the north or south. In a state where the agent has diverged (q_N or q_S for northward and southward diversion respectively), the agent always prefers to converge back to the path by moving in the opposite direction; failing this, the agent attempts to stay on that latitude (again, executing $\mathsf{stay}_{\mathsf{lat}}$).

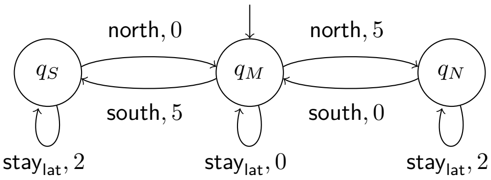

north, 0
north, 5
q_S
q_M
q_N
south, 5
south, 0
$\mathsf{stay}_{\mathsf{lat}}$, 2
$\mathsf{stay}_{\mathsf{lat}}$, 0
$\mathsf{stay}_{\mathsf{lat}}$, 2

Figure 2: A component modeling divergence from the path in two directions.

Since A_{move} and A_{diverge} share the same CAS and c-semiring, we can compose them to obtain $A_{\mathsf{move}} \bowtie A_{\mathsf{diverge}}$, a component that is aware of actions involved in patrolling and divergence, as well as their composed preferences. The resulting SCA is sketched in Figure 3; here, we see that the agent can be in one of six states, depending on its current direction and divergence. We now discuss the states whose outgoing transitions are drawn.

– When the agent is in state $q_{W,M}$, it is on the path and moving towards the western waypoint. We can see that the most preferred action is the self-transition labeled with the action $\mathsf{west}^\star$ (weight 0), which moves directly towards the western waypoint. Alternatively, the agent may diverge to the north- or southwest (weight 5), stay put (also weight 5) or turn around to the east (also weight 5). Failing that, the agent can opt to move due north or south (weight 10), and as a last resort the agent can opt to turn around to the southeast (weight 15).

– When in state $q_{E,S}$, the agent has diverged from the path to the south and is attempting to move to the eastern waypoint. Here, the most preferred action is $\mathsf{northeast}$, which brings the agent back on the path *and* moves towards the waypoint. Failing that, the agent can move due east (with weight 2), or return to the path while preserving its direction or turning around (weight 5). If those fail, the agent can stay in its current position (weight 7), or turn around while remaining diverged from the path (weight 7).

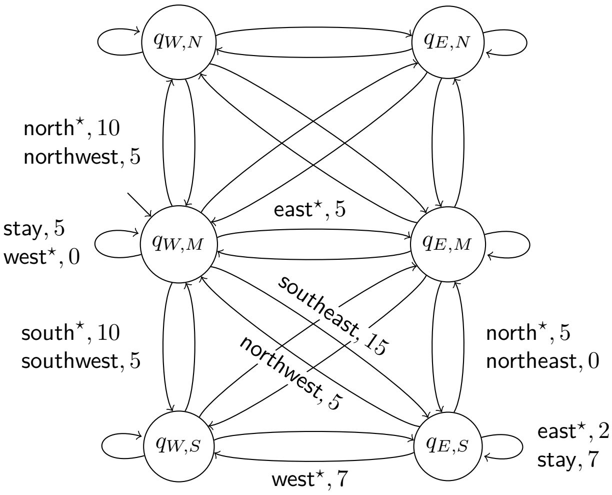


Figure 3: A sketch of $A_{\mathsf{move}} \bowtie A_{\mathsf{diverge}}$. For readability, we write $q_{x,y}$ instead of $\langle q_x, q_y \rangle$. To prevent clutter, only the transitions exiting $q_{W,M}$ and $q_{E,S}$ are drawn; all other transitions are similar.

We can conclude from the discussion above that our composition operator serves the purpose of letting the description of our final system remain concise; rather than manually specify the 6 states and 42 transitions of $A_{\mathsf{move}} \bowtie A_{\mathsf{diverge}}$, we specify the 5 states and 13 transitions of A_{move} and A_{diverge}. Moreover, since our CAS took care of which actions can compose meaningfully, and our c-semiring provided the composed preferences, there was no need to work these out manually.

One objection, however, is that some actions are insufficiently differentiated in terms of preference. Take, for example, the state $q_{W,M}$, where the actions $\mathsf{northwest}$, $\mathsf{southwest}$ and $\mathsf{east}^\star$ have the same weight; conceivably, we would prefer that the agent proceeds beside the path, rather than having it turn around, i.e., the agent should try to move around obstacles rather than switch target waypoints prematurely. Arguably, this problem is due to the simple reason that we really have two concerns at play here: A_{move} models the concern that the agent should patrol, while A_{diverge} models that the agent should stay on the path. By using the composition operator to compose A_{move} and A_{diverge}, we have lumped their preferences in the same c-semiring:

- The preference of 5 for $\mathsf{northwest}$ and $\mathsf{southwest}$ in $q_{W,M}$ is the result of the weight of 0 for west in q_W of A_{move}, as well as the weight 5 for north and south in q_M of A_{diverge}.
- The preference for 5 for $\mathsf{east}^\star$ in $q_{W,M}$ is the result of the weight 5 for east in q_W of A_{move} and 0 for $\mathsf{stay}_{\mathsf{lat}}$ in q_M of A_{diverge}.

To remedy this, we need to keep separation of concerns when composing SCAs. We first define how to move SCAs between c-semirings. Using functions between c-semirings, we can then define new composition operators. These operators first use a canonical injection to change the underlying c-semiring of the operands into a c-semiring composed of the underlying c-semirings of both operands. The results are then composed using the composition operator seen before. An added advantage is that we immediately obtain composition operators for SCAs with different underlying c-semirings.

Definition 13. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be an SCA and let $f : \mathbb{E} \rightarrow \mathbb{F}$ be a function. The application of f to A, written $f(A)$, is the SCA $\langle Q, \Sigma, \mathbb{F}, \rightarrow_h, q^0, f(t)\rangle$, where $\rightarrow_h$ is the smallest relation satisfying the rule*

$$\frac{q \xrightarrow{\sigma,\, e} q'}{q \xrightarrow{\sigma,\, f(e)}_h q'}$$

Definition 14. *Let $A_i = \langle Q, \Sigma, \mathbb{E}_i, \rightarrow_i, q_i^0, t_i\rangle$ be an SCA for $i \in \{0, 1\}$.*

- *The* product composition *of A_0 and A_1, written $A_0 \times A_1$, is given by*
$$\kappa_L^{\mathbb{E}_0 \times \mathbb{E}_1}(A_0) \bowtie \kappa_R^{\mathbb{E}_0 \times \mathbb{E}_1}(A_1)$$
- *If $\mathbb{E}_0$ is cancellative, then the* lexicographic composition *of A_0 and A_1, written $A_0 \rhd A_1$, is given by*
$$\kappa_L^{\mathbb{E}_0 \rhd \mathbb{E}_1}(A_0) \bowtie \kappa_R^{\mathbb{E}_0 \rhd \mathbb{E}_1}(A_1)$$
- *If $\mathbb{E}_0$ and $\mathbb{E}_1$ are cancellative, then the* join composition *of A_0 and A_1, written $A_0 \odot A_1$, is given by*
$$\kappa_L^{\mathbb{E}_0 \odot \mathbb{E}_1}(A_0) \bowtie \kappa_R^{\mathbb{E}_0 \odot \mathbb{E}_1}(A_1)$$

Using our new composition operators, we can can remedy the objection raised to the example of the patrolling agent. Specifically, the SCA $A_{\mathsf{move}} \rhd A_{\mathsf{diverge}}$ has transitions where the preference originating from A_{move} is present on the most significant component; accordingly, the transitions labeled with $\mathsf{northwest}$ and $\mathsf{southwest}$ in $q_{W,M}$ now have the weight $\langle 0, 5\rangle \in \mathbb{W} \rhd \mathbb{W}$, which makes them preferred over the transition labeled with $\mathsf{east}^\star$ with weight $\langle 5, 0\rangle \in \mathbb{W} \rhd \mathbb{W}$.

Compromise and harmonization One important observation about SCAs is that, in a state, the most preferred actions need not be composable. As a result, the most preferred action in the composed state need not be the composition of the most preferred actions of the components. This property of SCAs represents that they are capable of *compromise*: if components disagree on the most-preferred action, their composition can have a most-preferred action which forms a middle ground between the preferences of both components. In general, compromise is not observed in compositions where all actions are pairwise composable; this is the case for the example with A_{move} and A_{diverge} above. However, if we adjust A_{diverge} to prefer converging straight back to the path over converging in its general direction (see Figure 4), then the most-preferred action west in q_W is incomposable with the most-preferred action $\mathsf{north}^\star$ in q_S. However, west in q_W *is* composable with north in q_S, the latter having preference 1. Consequently, the most-preferred action in $q_{W,S}$ of $A_{\mathsf{move}} \bowtie A'_{\mathsf{diverge}}$ is $\mathsf{northwest}$ with weight 1.

A concept closely related to compromise is *harmonization*: where compromise is the result of incomposability between actions, harmonization can be regarded as the result of incomposability between preferences. Harmonization occurs when the most-preferred actions are composable, but their preferences do not compose as preferably as the preferences of other actions. For instance, consider a component A_L with the UNIX-semiring as underlying c-semiring; this component may perform an action σ that requires reading privileges or an action τ that requires writing privileges (c.f. Figure 5a). We can see that in A_L, both σ and τ are most preferred actions, on account of their preferences being incomparable. Another component A_R has a single action ρ, requiring reading privileges, too (c.f. Figure 5b). Suppose both σ and τ are composable with ρ. The composition $A_L \bowtie A_R$ has a single most-preferred action $\sigma \,\Box\, \rho$; this action is preferred over the composed action $\tau \,\Box\, \rho$, since the latter has preference $\{\mathsf{R}, \mathsf{W}\}$, while the former has preference $\{\mathsf{R}\}$.

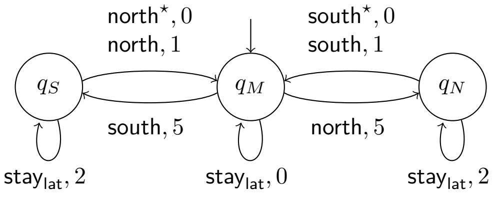
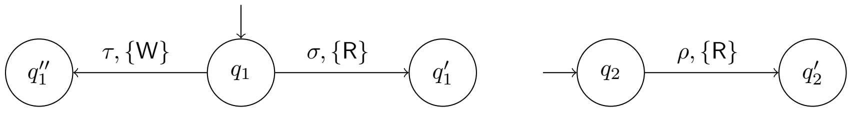


Figure 4: The adjusted divergence component A'_{diverge}.



(a) The component A_L.

(b) The component A_R.

Figure 5: Two components with the UNIX-semiring as underlying c-semiring.

5.2 Further observations

We conclude this section with some observations relevant to composition and homomorphisms. First, we note that homomorphisms are compatible with composition.

Lemma 10. *Let A_0 and A_1 be SCAs with underlying c-semiring $\mathbb{E}$, for $i \in \{0,1\}$. Let $h : \mathbb{E} \to \mathbb{F}$ be a homomorphism. Then $h(A_0 \bowtie A_1) = h(A_0) \bowtie h(A_1)$.*

Proof. The states of $h(A_0 \bowtie A_1)$ and $h(A_0) \bowtie h(A_1)$ are the same, as are the initial states, the underlying c-semiring and underlying CAS. Let t_i be the threshold value of A_i for $i \in \{0,1\}$. The threshold value of $h(A_0 \bowtie A_1)$ is $h(t_0 \otimes_{\mathbb{E}} t_2)$, which is equal to $h(t_1) \otimes_{\mathbb{F}} h(t_1)$ by the definition of homomorphism; the latter value is precisely the threshold value of $h(A_0) \bowtie h(A_1)$.

It remains to show that the transition relations of $h(A_0 \bowtie A_1)$ and $h(A_0) \bowtie h(A_1)$ are the same. If $\langle q_0, q_1\rangle \xrightarrow{\sigma,\, f} \langle q'_0, q'_1\rangle$ is a transition of $h(A_0 \bowtie A_1)$, then $\langle q_0, q_1\rangle \xrightarrow{\sigma_0 \,\square\, \sigma_1,\, e_0 \otimes_{\mathbb{E}} e_1} \langle q'_0, q'_1\rangle$ is a transition of $A_0 \bowtie A_1$ such that $q_i \xrightarrow{\sigma_i,\, e_i} q'_i$ is a transition of A_i for $i \in \{0,1\}$ with $h(e_0 \otimes_{\mathbb{E}} e_1) = f$ and $\sigma_0 \,\square\, \sigma_1 = \sigma$. But then $q_i \xrightarrow{\sigma_i,\, h(e_i)} q'_i$ is a transition of $h(A_i)$ for $i \in \{0,1\}$, making $\langle q_0, q_1\rangle \xrightarrow{\sigma_0 \,\square\, \sigma_1,\, h(e_0) \otimes_{\mathbb{F}} h(e_1)} \langle q'_0, q'_1\rangle$ a transition of $h(A_0) \bowtie h(A_1)$. Since $f = h(e_0 \otimes_{\mathbb{E}} e_1) = h(e_0) \otimes_{\mathbb{F}} h(e_1)$, this transition matches the transition we started with, making the transition relation of $h(A_0 \bowtie A_1)$ a subset of the transition relation of $h(A_0) \bowtie h(A_1)$. The proof of the other inclusion is similar. □

Next, we observe that $\bowtie$ is commutative and associative up to composability of SCAs. In accordance with the lemma below, we can drop parentheses when writing down compositions of SCAs.

Lemma 11. *Let R be a relation that relates SCAs with the same CAS and c-semiring. Then*

(a) The operator $\bowtie$ is commutative up to R.

(b) The operator $\bowtie$ is associative up to R.

Proof. Throughout this proof, let $A_i = \langle Q_i, \Sigma, \mathbb{E}, \to_i, q_i^0, t_i\rangle$ be an SCA for $i \in \{0,1,2\}$. It suffices to prove that $A_0 \bowtie A_1 = A_1 \bowtie A_0$ and $A_0 \bowtie (A_1 \bowtie A_2) = (A_0 \bowtie A_1) \bowtie A_2$, since $\bowtie$ preserves the CAS and c-semiring of its operands.

To see Lemma 11a, first observe that the underlying c-semiring and CAS of $A_0 \bowtie A_1$ and $A_1 \bowtie A_0$ are $\mathbb{E}$ and Σ respectively. The sets of states of $A_0 \bowtie A_1$ and $A_1 \bowtie A_0$ are $Q_0 \times Q_1$ and $Q_1 \times Q_0$, which we do not differentiate between (c.f. Section 3). Likewise, the initial states $\langle q_0^0, q_1^0\rangle$ and $\langle q_1^0, q_0^0\rangle$ are easily identified. Also, the threshold value of $A_0 \bowtie A_1$ is $t_0 \otimes t_1$, which is equal to the threshold value $t_1 \otimes t_0$ of $A_1 \bowtie A_0$.

It remains to show that $\langle q_0, q_1\rangle \xrightarrow{\sigma,\, e} \langle q'_0, q'_1\rangle$ is a transition of $A_0 \bowtie A_1$ if and only if $\langle q_1, q_0\rangle \xrightarrow{\sigma,\, e} \langle q'_1, q'_0\rangle$ is a transition of $A_1 \bowtie A_0$. Assuming the former, we know that there exist transitions $q_i \xrightarrow{\sigma_i,\, e_i}_i q'_i$ for $i \in \{0,1\}$ such that $\sigma_0 \bigcirc \sigma_1$, $\sigma_0 \,\square\, \sigma_1 = \sigma$ and $e_0 \otimes e_1 = e$. But then $A_1 \bowtie A_0$ has a transition $\langle q_1, q_0\rangle \xrightarrow{\sigma_1 \,\square\, \sigma_0,\, e_1 \otimes e_0} \langle q'_1, q'_0\rangle$. Since $\square$ and $\otimes$ are commutative, this transition is exactly the transition we set out to find. The proof in the other direction is analogous.

For Lemma 11b, first observe that the underlying c-semiring and CAS, as well as the set of states, initial state and threshold values of $(A_0 \bowtie A_1) \bowtie A_2$ and $A_0 \bowtie (A_1 \bowtie A_2)$ are identical by reasoning similar to the above. It remains to show that $\langle q_0, q_1, q_2\rangle \xrightarrow{\sigma,\, e} \langle q'_0, q'_1, q'_2\rangle$ is a transition of $(A_0 \bowtie A_1) \bowtie A_2$ if and only if it is a transition of $A_0 \bowtie (A_1 \bowtie A_2)$. Assuming the former, we know that there exist transitions $\langle q_0, q_1\rangle \xrightarrow{\sigma',\, e'} \langle q'_0, q'_1\rangle$ of $A_0 \bowtie A_1$ and $q_2 \xrightarrow{\sigma_2,\, e_2}_2 q'_2$ of A_2 such that $\sigma' \bigcirc \sigma_2$, $\sigma' \,\square\, \sigma_2 = \sigma$ and $e' \otimes e_2 = 3$; by the same argument, we derive that there exist transitions $q_i \xrightarrow{\sigma_i,\, e_i}_i q_i'$ of A_i for $i \in \{0,1,2\}$ such that $\sigma_0 \bigcirc \sigma_1$, $(\sigma_0 \square \sigma_1) \bigcirc \sigma_2$, $(\sigma_0 \square \sigma_1) \square \sigma_2 = \sigma$ and $(e_0 \otimes e_1) \otimes e_2 = e$. By associativity of $\square$ up to $\bigcirc$ and associativity of $\otimes$, we know that $\sigma_1 \bigcirc \sigma_2$, $\sigma_0 \bigcirc (\sigma_1 \square \sigma_2)$, $(\sigma_0 \square \sigma_1) \square \sigma_2 = \sigma_0 \square (\sigma_1 \square \sigma_2)$ and $(e_0 \otimes e_1) \otimes e_2 = e_0 \otimes (e_1 \otimes e_2)$. Accordingly, $\langle q_1, q_2 \rangle \xrightarrow{\sigma_1 \square \sigma_2,\, e_1 \otimes e_2} \langle q_1', q_2' \rangle$ is a transition of $A_1 \bowtie A_2$ and therefore $\langle q_0, q_1, q_2 \rangle \xrightarrow{\sigma,\, e} \langle q_0', q_1', q_2' \rangle$ is a transition of $A_0 \bowtie (A_1 \bowtie A_2)$. The proof in the other direction is analogous. □

Finally, we note that similar associativity and commutativity rules hold for the operators × and $\odot$, except that for commutativity, we need a (reflecting) homomorphism to move between orders of composition. Since this homomorphism does not change the order of preferences, we disregard it in further notation and treat × and $\odot$ as though they were commutative and associative, dropping parentheses whenever convenient. Lastly, $\triangleright$ is associative (as long as the most significant components have a cancellative underlying c-semiring), but not commutative, as one would expect from its definition.

Lemma 12. *Let $A_i = \langle Q_i, \Sigma, \mathbb{E}_i, \to_i, q_i^0, t_i \rangle$ be an SCA for $i \in \{0,1,2\}$. Then*

- *(a) There exists a reflecting homomorphism h such that $A_0 \times A_1 = h(A_1 \times A_0)$.*
- *(b) Also, $(A_0 \times A_1) \times A_2 = A_0 \times (A_1 \times A_2)$.*

Proof. For Lemma 12a, let $h : \mathbb{E}_1 \times \mathbb{E}_0 \to \mathbb{E}_0 \times \mathbb{E}_1$ be the function given for $e_1 \in \mathbb{E}_1$ and $e_0 \in \mathbb{E}_0$ by $h(e_1, e_0) = \langle e_0, e_1 \rangle$. It is easy to see that h is indeed a reflecting homomorphism. To see that $A_0 \times A_1 = h(A_1 \times A_1)$, first observe that $h \circ \kappa_L^{\mathbb{E}_1 \times \mathbb{E}_0} = \kappa_R^{\mathbb{E}_0 \times \mathbb{E}_1}$ and $h \circ \kappa_R^{\mathbb{E}_1 \times \mathbb{E}_0} = \kappa_L^{\mathbb{E}_0 \times \mathbb{E}_1}$ $(*)$. We can now derive as follows:

$$\begin{align*}
h(A_1 \times A_0) &= h\left(\kappa_L^{\mathbb{E}_1 \times \mathbb{E}_0}(A_1) \bowtie \kappa_R^{\mathbb{E}_1 \times \mathbb{E}_0}(A_0)\right) \tag{Definition 14}\\
&= h\left(\kappa_L^{\mathbb{E}_1 \times \mathbb{E}_0}(A_1)) \bowtie h(\kappa_R^{\mathbb{E}_1 \times \mathbb{E}_0}(A_0)\right) \tag{Lemma 10}\\
&= h\left(\kappa_R^{\mathbb{E}_1 \times \mathbb{E}_0}(A_0)) \bowtie h(\kappa_L^{\mathbb{E}_1 \times \mathbb{E}_0}(A_1)\right) \tag{Lemma 11a}\\
&= \kappa_L^{\mathbb{E}_0 \times \mathbb{E}_1}(A_0) \bowtie \kappa_R^{\mathbb{E}_0 \times \mathbb{E}_1}(A_1) \tag{By $(*)$}\\
&= A_0 \times A_1 \tag{Definition 14}
\end{align*}$$

For Lemma 12b, first observe that

$$\begin{align*}
\kappa_L^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2} \circ \kappa_L^{\mathbb{E}_0 \times \mathbb{E}_1} &= \kappa_L^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)} \tag{i}\\
\kappa_R^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)} \circ \kappa_L^{\mathbb{E}_1 \times \mathbb{E}_2} &= \kappa_L^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2} \circ \kappa_R^{\mathbb{E}_0 \times \mathbb{E}_1} \tag{ii}\\
\kappa_R^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)} \circ \kappa_R^{\mathbb{E}_1 \times \mathbb{E}_2} &= \kappa_R^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2} \tag{iii}
\end{align*}$$

Now, we can derive

$$\begin{align*}
(A_0 \times A_1) \times A_2 &= \kappa_L^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2}\left(\kappa_L^{\mathbb{E}_0 \times \mathbb{E}_1}(A_0) \bowtie \kappa_R^{\mathbb{E}_0 \times \mathbb{E}_1}(A_1)\right) \bowtie \kappa_R^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2}(A_2) \tag{Definition 14}\\
&= \left(\kappa_L^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2} \circ \kappa_L^{\mathbb{E}_0 \times \mathbb{E}_1}(A_0) \bowtie \kappa_L^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2} \circ \kappa_R^{\mathbb{E}_0 \times \mathbb{E}_1}(A_1)\right) \bowtie \kappa_R^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2}(A_2) \tag{Lemma 10}\\
&= \left(\kappa_L^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)}(A_0) \bowtie \kappa_L^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2} \circ \kappa_R^{\mathbb{E}_0 \times \mathbb{E}_1}(A_1)\right) \bowtie \kappa_R^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2}(A_2) \tag{By (i)}\\
&= \kappa_L^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)}(A_0) \bowtie \left(\kappa_L^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2} \circ \kappa_R^{\mathbb{E}_0 \times \mathbb{E}_1}(A_1) \bowtie \kappa_R^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2}(A_2)\right) \tag{Lemma 11b}\\
&= \kappa_L^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)}(A_0) \bowtie \left(\kappa_R^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)} \circ \kappa_L^{\mathbb{E}_1 \times \mathbb{E}_2}(A_1) \bowtie \kappa_R^{(\mathbb{E}_0 \times \mathbb{E}_1) \times \mathbb{E}_2}(A_2)\right) \tag{By (ii)}\\
&= \kappa_L^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)}(A_0) \bowtie \left(\kappa_R^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)} \circ \kappa_L^{\mathbb{E}_1 \times \mathbb{E}_2}(A_1) \bowtie \kappa_R^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)} \circ \kappa_R^{\mathbb{E}_1 \times \mathbb{E}_2}(A_2)\right) \tag{By (iii)}\\
&= \kappa_L^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)}(A_0) \bowtie \kappa_R^{\mathbb{E}_0 \times (\mathbb{E}_1 \times \mathbb{E}_2)}\left(\kappa_L^{\mathbb{E}_1 \times \mathbb{E}_2}(A_1) \bowtie \kappa_R^{\mathbb{E}_1 \times \mathbb{E}_2}(A_2)\right) \tag{Lemma 10}\\
&= A_0 \times (A_1 \times A_2) \tag{Definition 14}
\end{align*}$$

We thus conclude that $(A_0 \times A_1) \times A_2 = A_0 \times (A_1 \times A_2)$. □

Lemma 13. *Let R be a relation on SCAs that relates two SCAs if and only if both have a cancellative c-semiring.*

- *(a) $\odot$ is commutative up to R, modulo a reflecting homomorphism.*
- *(b) $\odot$ is associative up to R.*

Proof. The proof is similar to that of Lemma 12, with the same homomorphism h. □

Lemma 14. *Let $A_i = \langle Q_i, \Sigma, \mathbb{E}_i, \to_i, q_i^0, t_i \rangle$ be an SCA for $i \in \{0,1,2\}$ such that $\mathbb{E}_0$ and $\mathbb{E}_1$ are cancellative. Then $(A_0 \triangleright A_1) \triangleright A_2 = A_0 \triangleright (A_1 \triangleright A_2)$*

Proof. The proof is similar to that of Lemma 12b. □

6 Linear Temporal Logic

In this section, we develop an operational model of SCAs, so as to characterise their *exhibited* behavior in terms of the (infinite) sequences of actions they allow, based on their threshold value. We show that this behavior can be captured by a Büchi-automaton. We then propose an instance of Linear Temporal Logic (LTL), called LTL_Σ, in which atomic propositions are the actions of a CAS Σ. We use LTL_Σ to describe the *desired* behavior of SCAs. We then show that a formula ϕ of LTL_Σ can be translated into a Büchi-automaton that accepts precisely the streams that validate ϕ. With this construction, we propose a decision procedure that uses the classic observations for Büchi-automata highlighted in Subsection 3.3, in the same fashion as [33]. If the behavior of an SCA $A = A_1 \bowtie A_1 \bowtie \ldots \bowtie A_n$ does *not* satisfy a formula ϕ, the supposed decision procedure provides us with a specific instance of the behavior of A that falsifies ϕ. The novelty of SCAs is that we can, in some circumstances, analyze the behavior to get an indication as to which components in $\{A_1, A_2, \ldots, A_n\}$ were responsible for letting this behavior exist in A.

6.1 Operational model

Our description of the behavior of an SCA is based on the assumption that SCAs are allowed to execute only actions whose preference is bounded from below by the threshold value of the SCA. As such, the threshold value is treated as a *minimum satisfaction level* for the component. By adjusting the threshold value, the SCA can tune its standards: with a high threshold value, the SCA expects to be able to execute high-preference actions, while a lower threshold value indicates that the SCA is content with lower-preference actions as well.

Definition 15. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA. The* trace relation *of A, written $\rightleftharpoons_A$, and its* infinitary lifting, *written $\rightleftharpoons_A^\omega$, are the smallest relations on $(Q \times \Sigma)^\omega$ satisfying the rule*

$$\frac{\langle \mu, \nu \rangle \in Q^\omega \times \Sigma^\omega \qquad e \in \mathbb{E} \qquad t \leq e \qquad \mu(0) \xrightarrow{\nu(0),\, e} \mu(1)}{\mu \rightleftharpoons_A \nu} \qquad \frac{\forall n \in \mathbb{N}.\ \mu^{(n)} \rightleftharpoons_A \nu^{(n)}}{\mu \rightleftharpoons_A^\omega \nu}$$

The language *of A, written $L(A)$, is the smallest set satisfying the rule*

$$\frac{\mu \rightleftharpoons_A^\omega \nu \qquad \mu(0) = q^0}{\nu \in L(A)}$$

Intuitively, the language of an SCA A includes precisely the streams of actions ν, such that we can find a stream of states μ in A, starting in the initial state, where the actions on the transitions between the states are labeled with the actions from ν, and with preference values bound from below by the threshold value. When A is an SCA and $\nu \in L(A)$, we refer to ν as a *behavior* of A.

6.2 SCAs to Büchi-automata

The language of an SCA is preserved by some c-semiring maps, as shown below.

Lemma 15. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA, and let $h : \mathbb{E} \to \mathbb{F}$ be a function. If h is t-reflecting, then A and $h(A)$ share their behaviors, i.e., $L(A) = L(h(A))$.*

Proof. Assume that h is t-reflecting. We first show that $\rightleftharpoons_A$ coincides with $\rightleftharpoons_{h(A)}$, and $\rightleftharpoons_A^\omega$ with $\rightleftharpoons_{h(A)}^\omega$. Assume that $\langle \mu, \nu \rangle \in (Q \times \Sigma)^\omega$ is such that $\mu \rightleftharpoons_A \nu$ holds. This is true precisely when there exists a transition $\mu(0) \xrightarrow{\nu(0),\, e} \mu(1)$ in A such that $t \leq e$, which in turn exists if and only if there is a transition $\mu(0) \xrightarrow{\nu(0),\, h(e)} \mu(1)$ in $h(A)$, and $h(t) \leq h(e)$. Since $h(t)$ is the threshold value of A, the latter holds if and only if $\mu \rightleftharpoons_{h(A)} \nu$. Now, $\mu \rightleftharpoons_A^\omega \nu$ if and only if $\mu^{(n)} \rightleftharpoons_A \nu^{(n)}$ for all $n \in \mathbb{N}$, if and only if $\mu^{(n)} \rightleftharpoons_{h(A)} \nu^{(n)}$ for all $n \in \mathbb{N}$, if and only if $\mu \rightleftharpoons_{h(A)}^\omega \nu$. Let $\nu \in L(A)$; by definition this is true if there exists a $\mu \in Q^\omega$ such that $\nu(0) = q^0$ and $\mu \rightleftharpoons_A^\omega \nu$. But due to the above, the latter holds if and only if $\mu \rightleftharpoons_{h(A)}^\omega \nu$. The preceding is necessary and sufficient to conclude that $\nu \in L(h(A))$. □

We can use Lemma 15 to show that languages recognized by SCAs are also recognized by Büchi-automata.

Lemma 16. *Let A be an SCA. Then there exists a Büchi-automaton A_B such that $L(A) = L(A_B)$.*

Proof. Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$. First, note that $\mathbf{t}_t$ is t-reflecting (by Lemma 26). Then $L(\mathbf{t}_t(A)) = L(A)$ by Lemma 15. We construct a Büchi-automaton A_B as follows:

- The set of states is Q, the set of states of $\mathbf{t}_t(A)$, and the initial state is q^0.
- The alphabet is Σ, the carrier of the CAS of $\mathbf{t}_t(A)$.

- The set of accepting states is again Q, i.e., all states of our Büchi-automaton are accepting.
- The transition relation is the smallest relation $\rightarrow_B$ that satisfies

$$\frac{q \xrightarrow{\sigma,\, \top}_{\mathbf{t}_t} q'}{q \xrightarrow{\sigma}_B q'}$$

It now suffices to prove that $L(A_B) = L(\mathbf{t}_t(A))$. As in Lemma 15, we first set out to prove that $\rightleftharpoons_{A_B}$ coincides with $\rightleftharpoons_{\mathbf{t}_t(A)}$. Let $\langle \mu, \nu \rangle \in (Q \times \Sigma)^\omega$ be such that $\mu \rightleftharpoons_{A_B} \nu$. This is true if and only if there exists a transition $\mu(0) \xrightarrow{\nu(0)}_B \mu(1)$, which (by definition of $\rightarrow_B$) is true if and only if there exists a transition $\mu(0) \xrightarrow{\nu(0),\, \top}_{\mathbf{t}_t} \mu(1)$. Since $\mathbf{t}_t(t) = \top \leq_{\mathbb{B}} \top$, the latter holds if and only if $\mu \rightleftharpoons_{\mathbf{t}_t(A)} \nu$. By a similar argument as in Lemma 15, we find that $\mu \rightleftharpoons^\omega_{A_B} \nu$ if and only if $\mu \rightleftharpoons^\omega_{\mathbf{t}_t(A)} \nu$.

Now, let $\nu \in L(A_B)$; this is true if and only if there exists a stream $\mu \in Q^\omega$ such that $\mu(0) = q^0$, $\mu(n) \in Q$ for infinitely many $n \in \mathbb{N}$ (this claim holds vacuously) and $\mu \rightleftharpoons^\omega_{A_B} \nu$. By the reasoning above, the latter holds if and only if $\mu \rightleftharpoons^\omega_{\mathbf{t}_t(A)} \nu$, which is true precisely when $\nu \in L(\mathbf{t}_t(A)) = L(A)$; we conclude that $L(A_B) = L(A)$. □

6.3 Desired behavior

To describe the desired behavior of the SCA, we discuss a modest variation of Linear Temporal Logic [33, 24] that is capable of recognizing composed actions by their component actions. In this logic, every formula describes one or more behaviors, as is made precise in the definitions that follow.

Definition 16. *Let Σ be a CAS. The set of valid formulas of LTL$_\Sigma$, written $\mathcal{L}_\Sigma$, is the smallest set satisfying*

$$\frac{}{\top \in \mathcal{L}_\Sigma} \qquad \frac{\sigma \in \Sigma}{\sigma \in \mathcal{L}_\Sigma} \qquad \frac{\phi \in \mathcal{L}_\Sigma}{\neg\phi \in \mathcal{L}_\Sigma} \qquad \frac{\phi, \psi \in \mathcal{L}_\Sigma}{\phi \wedge \psi \in \mathcal{L}_\Sigma \qquad \phi \, U \, \psi \in \mathcal{L}_\Sigma} \qquad \frac{\phi \in \mathcal{L}_\Sigma}{\phi^{\mathsf{Y}} \in \mathcal{L}_\Sigma}$$

When writing down formulas of LTL$_\Sigma$, the order of precedence is as follows: first comes the right-side denoted unary connective Y, followed by the left-side denoted unary connective $\neg$, and last come the binary connectives $\wedge$ and U. The formula $\neg\phi^{\mathsf{Y}} \wedge \psi$ should thus be read as $(\neg(\phi^{\mathsf{Y}})) \wedge \psi$. For some formulas we need to use parentheses to disambiguate, for example between $\phi \wedge (\psi \, U \, \chi)$ and $(\phi \wedge \psi) \, U \, \chi$.

Definition 17. *The* semantics *of LTL$_\Sigma$, written $\models$, is the smallest relation between $\mathcal{L}_\Sigma$ and Σ^ω satisfying*

$$\frac{\nu \in \Sigma^\omega}{\nu \models \top} \qquad \frac{\nu \in \Sigma^\omega \qquad \nu(0) = \sigma}{\nu \models \sigma} \qquad \frac{\nu \not\models \phi}{\nu \models \neg\phi} \qquad \frac{\phi \models \nu \text{ and } \psi \models \nu}{\nu \models \phi \wedge \psi}$$

$$\frac{\langle \nu, \phi \rangle \in \Sigma^\omega \times \mathcal{L}_\Sigma \qquad \exists n \in \mathbb{N}.\left[\nu^{(n)} \models \psi \text{ and } \forall k < n. \nu^{(k)} \models \phi\right]}{\nu \models \phi \, U \, \psi}$$

$$\frac{\phi \in \mathcal{L}_\Sigma \qquad \nu \models \phi \qquad \lambda, \xi \in \Sigma^\omega \qquad \forall n \in \mathbb{N}.\ \nu(n) = \lambda(n) \Box \xi(n)}{\lambda \models \phi^{\mathsf{Y}}}$$

We extend the use of $\models$ to SCAs as follows: if A is an SCA with underlying CAS Σ and $\phi \in \mathcal{L}_\Sigma$, then $A \models \phi$ if and only if for all $\nu \in L(A)$ it holds that $\nu \models \phi$.

The symbol $\top$ is thus interpreted as a formula that holds for every behavior, while $\sigma \in \Sigma$ holds for all behaviors whose head is σ. The formula $\phi \wedge \psi$ holds for all behaviors that satisfy both ϕ and ψ, while the formula $\neg\phi$ holds for the behaviors that do not satisfy ϕ. Lastly, the formula $\phi \, U \, \psi$ asserts that there exists an $n \in \mathbb{N}$ such that all derivatives up to the n-th derivative of the behavior satisfy ϕ, and that the n-th derivative satisfies ψ; this connective is therefore best read as *ϕ holds for a finite number of steps, after which ψ holds*, or, more informally, *ϕ holds until ψ does*.

The unary connective Y is novel and deserves some further explanation. Inspecting the definition, we can see that ϕ^{Y} is validated by behaviors for which there exists another behavior, such that the composed behavior of the two validates ϕ. Note that $\nu \models \phi$ implies $\nu \models \phi^{\mathsf{Y}}$, since $\Box$ is idempotent. Also, $\nu \models \sigma^{\mathsf{Y}}$ holds for behaviors ν whose head $\nu(0)$ is an action that has σ as a component action.

Expressiveness Using the rather minimal syntax introduced in Definition 16, one can derive a number of useful connectives and modalities, which give a nice syntactic sugar over the language and enrich its intuitive expressiveness. Following [2], we mention a few:

- The classic propositional directives of $\vee$ (disjunction), $\rightarrow$ (implication) and $\leftrightarrow$ (equivalence) can be derived using the connectives $\neg$ and $\wedge$; for example, $\phi \vee \psi$ can be defined as $\neg(\neg\phi \wedge \neg\psi)$, by De Morgan's law.

- The modality $\Diamond\phi$, defined as a shorthand for $\top\, U\, \phi$. By the semantics in Definition 17, we can see that this formula holds for behaviors $\nu \in \Sigma^\omega$ where there exists an $n \in \mathbb{N}$ such that $\top \models \nu^{(k)}$ for all $k < n$, and $\phi \models \nu^{(n)}$. Since the former claim is vacuously true, we can interpret $\Diamond\phi$ to be a formula that holds for all $\nu \in \Sigma^\omega$ where ϕ holds *after a finite number of steps*. Informally, we can thus read this formula as *eventually ϕ holds*.
- The dual modality to the above, $\Box\phi$, defined as $\neg\Diamond\neg\phi$. From this definition, we can see that this formula holds for a behavior $\nu \in \Sigma^\omega$ such that there is *not* a finite number of steps $n \in \mathbb{N}$ for which $\phi \models \nu^{(n)}$ does *not* hold. This implies that $\phi \models \nu^{(n)}$ must hold for all $n \in \mathbb{N}$. We can therefore read $\Box\phi$ as *ϕ always holds*.

As an example of the expressiveness of LTL for SCAs, consider the SCA $A = A_{\mathsf{move}} \bowtie A_{\mathsf{diverge}}$ as seen in Section 5. If we want to verify that the agent can keep patrolling indefinitely, we can check whether $A \models \Box(\Diamond\mathsf{east}^{\curlyvee} \wedge \Diamond\mathsf{west}^{\curlyvee})$ holds; if it does, then any behavior will, at any point, eventually perform an action composed of going east, and will also eventually perform an action composed of going west. Note that the connective $\curlyvee$ here allows us to also capture the behaviors where the agent turns around to the east by, for example, performing the action $\mathsf{southeast}$.

6.4 Towards model checking

We now show that, given a formula $\phi \in \mathcal{L}_\Sigma$, we can construct a Büchi-automaton A that accepts precisely the streams that are modeled by ϕ. In [33], Vardi claims that the structure of the proof of the lemma below appears in [29]; to our eyes, the latter paper appears to be concerned with a different type of logic, so we give a full proof instead of a citation. The part concerned with the $\curlyvee$-operator is novel (if somewhat obvious).

Lemma 17. *Let $\phi \in \mathcal{L}_\Sigma$. Then we can construct a Büchi-automaton such that $\nu \models \phi$ if and only if $\nu \in L(A)$.*

Proof. We construct the desired Büchi-automaton by induction on the structure of ϕ. For the base case, we have that either $\phi = \top$ or $\phi = \sigma$ for some $\sigma \in \Sigma$. In either case, it should be clear to construct a Büchi-automaton such that $\nu \models \phi$ if and only if $\nu \in L(A)$. For the inductive step, assume that the claim holds for all subformulas of ϕ; we distinguish based on the operator at hand:

- If $\phi = \neg\phi_0$, by the induction hypothesis there exists a Büchi-automaton A_{ϕ_0} such that $\nu \models \phi_0$ if and only if $\nu \in L(A_{\phi_0})$. By Lemma 7, we can construct a Büchi-automaton A_ϕ such that $\nu \in L(A_\phi)$ if and only if $\nu \notin L(A_{\phi_0})$. By construction, we now know that $\nu \in L(A_\phi)$ if and only if $\nu \not\models \phi_0$, i.e., $\nu \models \neg\phi_0 = \phi$.
- If $\phi = \phi_0 \wedge \phi_1$, by the induction hypothesis there exist Büchi-automata A_i for $i \in \{0, 1\}$ such that $\nu \in L(A_i)$ if and only if $\nu \models \phi_i$ for $i \in \{0, 1\}$. By Lemma 6, we can construct a Büchi-automaton A such that $L(A) = L(A_0) \cap L(A_1)$. Now $\nu \in L(A)$ if and only if $\nu \in L(A_i)$ for $i \in \{0, 1\}$ if and only if $\nu \models \phi_i$ for $i \in \{0, 1\}$ if and only if $\nu \models \phi_0 \wedge \phi_1 = \phi$.
- If $\phi = \phi_0\, U\, \phi_1$, by the induction hypothesis there exist Büchi-automata A_i for $i \in \{0, 1\}$ such that $\nu \in L(A_i)$ if and only if $\nu \models \phi_i$ for $i \in \{0, 1\}$. By Lemma 8, we can construct a Büchi-automaton A such that $\nu \in L(A)$ if and only if there exists an $n \in \mathbb{N}$ such that for all $0 \leq k < n$ it holds that $\nu^{(k)} \in L(A_0)$ and $\nu^{(n)} \in L(A_1)$. But the latter condition holds if and only if there exists an $n \in \mathbb{N}$ such that for all $0 \leq k < n$ it holds that $\nu^{(k)} \models \phi_0$ and $\nu^{(n)} \in \phi_1$, i.e., precisely when $\nu \models \phi_0\, U\, \phi_1 = \phi$.
- If $\phi = \phi_0^{\curlyvee}$, by the induction hypothesis there exists a Büchi-automaton $A_0 = \langle Q_0, \Delta, \rightarrow_{\phi_0}, q^0, F\rangle$ such that $\nu \in L(A_0)$ if and only if $\nu \models \phi_0$. Now, let A_ϕ be the Büchi-automaton $\langle Q, \Delta, \rightarrow_\phi, q^0, F\rangle$ be the Büchi-automaton where $\rightarrow_\phi$ is the smallest relation satisfying the rule

  $$\frac{q \xrightarrow{\sigma}_\psi q' \qquad \sigma = \rho \,\Box\, \tau}{q \xrightarrow{\rho}_\phi q'}$$

  If $\nu \in L(A_\phi)$, then there exists a $\mu \in Q^\omega$ such that $\mu \rightleftharpoons^\omega_{A_\phi} \nu$. Thus, for every $n \in \mathbb{N}$, we have that $\mu(n) \xrightarrow{\nu(n)}_\phi \mu(n+1)$. But then for every $n \in \mathbb{N}$ there exist σ_n, τ_n such that $\sigma_n = \nu(n) \,\Box\, \tau_n$ and $q \xrightarrow{\sigma_n}_{\phi_0} q'$, by construction of $\rightarrow_\phi$. Choose $\lambda(n) = \sigma_n$ and $\xi(n) = \tau_n$. We then know that $\mu(n) \xrightarrow{\lambda(n)}_{\phi_0} \mu(n+1)$ for all $n \in \mathbb{N}$, thus $\mu \rightleftharpoons^\omega_{A_{\phi_0}} \lambda$ and therefore $\lambda \in L(A_{\phi_0})$. Since $\lambda \models \phi_0$ and $\lambda(n) = \nu(n) \,\Box\, \xi(n)$, also $\nu \models {\phi_0}^{\curlyvee} = \phi$.

  Conversely, let $\nu \models \phi = \phi_0^{\curlyvee}$. Then there exist $\lambda, \xi \in \Sigma^\omega$ such that $\lambda \models \phi_0$ and $\lambda(n) = \nu(n) \,\Box\, \xi(n)$. Then $\lambda \in L(A_{\phi_0})$, thus there exists a $\mu \in Q^\omega$ such that $\mu \rightleftharpoons^\omega_{A_{\phi_0}} \lambda$, $\mu(0) = q^0$ and $\mu^{-1}(F)$ is infinite. Since for every $n \in \mathbb{N}$ we have that $\mu(n) \xrightarrow{\lambda(n)}_{\phi_0} \mu(n+1)$, also $\mu(n) \xrightarrow{\nu(n)}_\phi \mu(n+1)$. As a consequence, $\mu \rightleftharpoons^\omega_{A_\phi} \nu$, thus $\nu \in L(A_\phi)$. □

Unfortunately, the steps used for the negation and U-operator in the proof above can cause an exponential blowup in the number of states of the automaton being constructed. As a result, the worst-case complexity of the construction is *non-elementary* [33], i.e., given by a nested series of exponentials, whose depth depends on the depth of ϕ.

For an approach that is computationally feasible, we suspect that it is possible to extend the construction found in [33] to work with the $\curlyvee$-operator. The problem here is that it is not immediately obvious how to obtain the negated dual of a formula $\phi^{\curlyvee}$, since $\neg\phi^{\curlyvee}$ is generally not equivalent to $(\neg\phi)^{\curlyvee}$. Consider for example the case where $\Sigma = \mathcal{M}$, and set $\phi = \mathsf{east}$; if $\nu \in \mathcal{M}^\omega$ such that $\nu(0) = \mathsf{east}$, we find that $\nu \not\models \neg\mathsf{east}^{\curlyvee}$, while $\nu \models (\neg\mathsf{east})^{\curlyvee}$.

Decision procedure We now combine the material in preceding paragraphs to sketch a procedure that decides whether $A \models \phi$. This procedure follows the general setup of [33].

- Given an SCA A, construct the Büchi-automaton $L(A_B)$ such that $L(A) = L(A_B)$ (c.f. Lemma 16).
- Given a formula $\phi \in \mathcal{L}_\Sigma$, obtain the Büchi-automaton $L(A_\phi)$ such that $\nu \models \phi$ if and only if $\nu \in L(A_\phi)$.
- Construct the Büchi-automaton $\overline{A_\phi}$ such that $\nu \in L(\overline{A_\phi})$ if and only if $\nu \notin L(A_\phi)$ (c.f. Lemma 7).
- Construct the Büchi-automaton $A_\cap$ such that $L(A_\cap) = L(A_B) \cap L(\overline{A_\phi})$ (c.f. Lemma 6).
- Check whether $L(A_\cap)$ is empty (using Lemma 5); if so, output that $A \models \phi$, otherwise output that $A \not\models \phi$.

For correctness of the above, suppose that $L(A_\cap)$ is non-empty. This is true if and only if there exists a $\nu \in \Sigma^\omega$ such that $\nu \in L(A_B)$ and $\nu \in L(\overline{A_\phi})$. But this holds if and only if $\nu \in L(A)$ and $\nu \notin L(A_\phi)$, which in turn is true if and only if $\nu \in L(A)$ and $\nu \not\models \phi$, i.e., when $A \not\models \phi$.

6.5 Diagnostics

If A is an SCA with $\mu \rightleftharpoons^\omega_A \nu$, we know that μ induces an infinite path through Q; furthermore, this path is not only labeled with actions, but also with preferences. The definition below provides us with a summary of the preferences that appear along μ, which we call the *diagnostic value*. Note that, since there may be more than one preference for the action $\nu(n)$ from state $\mu(n)$ to state $\mu(n+1)$, we need to take some care in condensing the values.

Definition 18. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0 \rangle$ be an SCA. If $\mu \rightleftharpoons^\omega_A \nu$, we call $\eta \in \mathbb{E}^\omega$ a* diagnostic stream *of $\langle \mu, \nu \rangle$, when we have that $\mu(n) \xrightarrow{\nu(n),\, \eta(n)} \mu(n+1)$ for all $n \in \mathbb{N}$. We write $H_{\mu,\nu}$ for the* set of diagnostic streams *of $\langle \mu, \nu \rangle$. The* diagnostic value *of $\langle \mu, \nu \rangle$, written $e_{\mu,\nu}$, is then defined as*

$$e_{\mu,\nu} = \bigwedge \left\{ \bigvee \{\eta'(k) : \eta' \in H_{\mu,\nu}\} : k \in \mathbb{N} \right\}$$

The diagnostic value gives us a necessary condition for $\mu \rightleftharpoons^\omega_A \nu$ to hold.

Lemma 18. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA. If $\mu \rightleftharpoons^\omega_A \nu$, then $t \leq e_{\mu,\nu}$.*

Proof. We know that $\mu \rightleftharpoons^\omega_A \nu$, i.e., $\mu(n) \xrightarrow{\nu(n),\, e_n} \mu(n+1)$ for all $n \in \mathbb{N}$. Define $\eta \in \mathbb{E}^\omega$ as $\eta(n) = e_n$. Then η is a diagnostic stream of $\langle \mu, \nu \rangle$, by construction. Thus $\eta \in H_{\mu,\nu}$, and moreover, $t \leq \eta(n)$ for all $n \in \mathbb{N}$. Since $\eta(n) \leq \bigvee\{\eta'(n) : \eta' \in H_{\mu,\nu}\}$ for all $n \in \mathbb{N}$, it follows that $t \leq \bigvee\{\eta'(n) : \eta' \in H_{\mu,\nu}\}$ for all $n \in \mathbb{N}$. Therefore, $t \leq \bigwedge\{\bigvee\{\eta'(n) : \eta' \in H_{\mu,\nu}\}\} = e_{\mu,\nu}$. □

By Lemma 18, if for an SCA $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ it holds that $\nu \in L(A)$, and we want to adjust A such that $\nu \notin A$, we can do so by adjusting the threshold value t such that $t \not\leq e_{\mu,\nu}$ for all μ with $\mu \rightleftharpoons^\omega_A \nu$. We note that, as $\rightarrow$ is finite, there are only finitely many distinct such $e_{\mu,\nu}$, and so this transformation can be applied iteratively. A caveat to this method is that, by adjusting the threshold value of A, we may also eliminate other (possibly desirable) behavior from A; in the worst case, it may turn out that $L(A) = \emptyset$ after adjusting t. Another caveat is that if $e_{\mu,\nu} = \mathbf{1}$ for some μ with $\mu \rightleftharpoons^\omega_A \nu$, then such a t does not exist. This, however, is to be expected: if $e_{\mu,\nu} = \mathbf{1}$, then ν is, in some intuitive sense, part of the "optimal" behavior of A; if the optimal behavior of an SCA is undesirable, then something must be wrong with its preferences!

A different use of Lemma 18 allows us to find out which components of a composed SCA can be regarded as "responsible" for allowing some undesired behavior ν. Suppose, for instance, that $A = \bowtie_{i \in I} A_i$, i.e., A is the composition of SCAs $A_i = \langle Q_i, \Sigma, \mathbb{E}, \rightarrow_i, q_i^0 \rangle$ for $i \in I$. Then, if $\nu \in L(A)$ is some behavior exhibited by A, by Lemma 18, it holds that $t \leq e_{\mu,\nu}$ for some $\mu \in Q^\omega$. In particular, it holds that $\bigotimes_{i \in I} t_i \leq e_{\mu,\nu}$.

We adopt the convention that, if $I' \subseteq I$ such that $\bigotimes_{i \in I'} t_i \leq e_{\mu,\nu}$, we call I' a *suspect* subset of I. We can now see that if I' is a suspect subset of I, we need to change at least one t_i with $i \in I'$ if we want $\mu \not\rightleftharpoons^\omega_A \nu$ to hold — if not, then $\bigotimes_{i \in I'} t_i \leq e_{\mu,\nu}$, thus $\bigotimes_{i \in I} t_i = \bigotimes_{i \in I'} t_i \otimes \bigotimes_{i \in I \setminus I'} t_i \leq e_{\mu,\nu}$ by intensivity of $\otimes$ (Lemma 3). By extension, we can see that we need to adjust at least as many threshold values as there are mutually disjoint suspect subsets of I — i.e., if $\mathcal{I} = \{I_1, I_2, \ldots, I_k\} \subseteq 2^I$ such that I_i is suspect for $1 \leq i \leq k$ and $I_i \cap I_j = \emptyset$ for $1 \leq i < j \leq k$, then A has at least $|\mathcal{I}|$ components that need adjusting. Furthermore, if I' is a suspect subset of

I and does not have a strict subset that is suspect, we call I' *culpable*. If I' is culpable, then A_i with $i \in I'$ can be said to contribute towards allowing the behavior ν to exist, or (at least), not prevent it.

Using the above, if $\phi \in \mathcal{L}_\Sigma$ is a formula of LTL_Σ such that $A \not\models \phi$, then with the decision procedure outlined above we obtain (through Lemmas 5 and 16) streams $\mu \in Q^\omega$ and $\nu \in \Sigma^\omega$ such that $\nu \not\models \phi$, $\nu \in L(A)$ and $\mu \rightleftharpoons_A^\omega \nu$. Moreover, μ and ν have a simple description (also by Lemma 5), thus we can compute $e_{\mu,\nu}$. Using the discussion above, we can then identify the culpable components of A, i.e., the components of A responsible for allowing the behavior ν to exist in A.

Example Suppose that our agent is modeled by the automaton $A_{\mathsf{move}} \triangleright A_{\mathsf{diverge}}$, and suppose that the threshold value of both automata is $5 \in \mathbb{W}$. If we want to check that the agent never strays from the path to the north, we can do so by verifying whether $A \models \neg\Diamond\mathsf{north}^{\curlyvee}$ holds. This is decidable, and if the outcome is negative, we obtain a behavior ν for which $\nu \models \neg\neg\Diamond\mathsf{north}^{\curlyvee}$ holds, i.e., $\nu \models \Diamond\mathsf{north}^{\curlyvee}$ holds.

It turns out that such a behavior exists; consider, for instance, the behavior ν, with $\nu(0) = \mathsf{northwest}$ and $\nu(n) = \mathsf{west}$ for $n \geq 1$. The accompanying stream of states for this behavior is μ, with $\mu(0) = q_{W,M}$ and $\mu(n) = q_{W,N}$ for $n \geq 1$. The unique diagnostic stream of $\langle \mu, \nu \rangle$ is η, with $\eta(0) = \langle 0, 5 \rangle$ and $\eta(n) = \langle 0, 2 \rangle$ for $n \geq 1$; hence, the diagnostic value is $e_{\mu,\nu} = \langle 0, 5 \rangle$.

Now, $A_{\mathsf{move}} \triangleright A_{\mathsf{diverge}} = \kappa_L^{\mathbb{W} \triangleright \mathbb{W}}(A_{\mathsf{move}}) \bowtie \kappa_R^{\mathbb{W} \triangleright \mathbb{W}}(A_{\mathsf{diverge}})$. Also, the threshold value of $\kappa_L^{\mathbb{W} \triangleright \mathbb{W}}(A_{\mathsf{move}})$ is $\langle 5, 0 \rangle$ while the threshold value of $\kappa_R^{\mathbb{W} \triangleright \mathbb{W}}(A_{\mathsf{diverge}})$ is $\langle 0, 5 \rangle$. Since $\langle 5, 0 \rangle \not\leq_{\mathbb{W} \triangleright \mathbb{W}} \langle 0, 5 \rangle$, while $\langle 0, 5 \rangle \leq_{\mathbb{W} \triangleright \mathbb{W}} \langle 0, 5 \rangle$, we find that $\{A_{\mathsf{diverge}}\}$ is the only culpable subset of components. To eliminate ν from $A_{\mathsf{move}} \triangleright A_{\mathsf{diverge}}$, we thus need to choose a threshold value t for A_{diverge} such that $t \not\leq_{\mathbb{W} \triangleright \mathbb{W}} e_{\mu,\nu}$; for example, $t = \langle 0, 2 \rangle$ would do.

7 Propositional Dynamic Logic

In this section, we propose an alternative logic for SCAs inspired by Propositional Dynamic Logic (PDL), called PDL_Σ. This logic contrasts LTL_Σ in the previous section in that for LTL_Σ our formulas pertained to actions with preferences above the threshold value, whereas for PDL_Σ our formulas are only concerned with the *optimal* behavior (for a certain notion of optimality). Furthermore, whereas LTL_Σ makes assertions about *infinite* sequences (i.e., streams) of actions, PDL_Σ is concerned with *finite* sequences (i.e., words) of actions.

First, we define the notion of *behavior* used for this logic, specifically the *optimal behavior*, parameterised in terms of a partial order. We then carry on by defining the syntax and semantics of PDL_Σ, along with a discussion on the intuition behind the connectives and a small example of its use. Subsequently, two non-trivial partial orders are proposed, both of which include the possibility of the agent *idling*, so as to compare the merit of *doing something* versus *not doing something.* We conclude this section with some further investigation into the structure of preferences of optimal behavior, with the partial order instantiated to one of the proposed orders.

For the purpose of the discussion ahead, it is useful to make a distinction between two types of words. We refer to a finite sequence of actions from a CAS as an *action word*, whereas a finite sequence of preferences from a c-semiring is a *preference word.*

7.1 Optimal behavior

We first need to formally establish what we mean by *(optimal) behavior.* Suppose our agent is modeled by an SCA A, and is currently in a state $q \in Q$. The behavior exhibited in this state is determined by all series of transitions that start in q, where each preference should be feasible (i.e., should not be the bottom preference).

Definition 19. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be an SCA. The* behavior relation *of A, written $\Rightarrow_A$, is the smallest relation in $Q \times \Sigma^* \times \mathbb{E}^* \times Q$ satisfying the rules*

$$\frac{}{q \xRightarrow{\epsilon,\ \epsilon}_A q} \qquad \frac{q \xrightarrow{\sigma, e} q' \qquad e \neq \mathbf{0} \qquad q' \xRightarrow{w,\ x}_A q''}{q \xRightarrow{\sigma\cdot w,\ e\cdot x}_A q''}$$

in which we write $q \xRightarrow{w,\ x}_A q'$ for $\langle q, w, x, q'\rangle \in \Rightarrow_A$. If $q \xRightarrow{w,\ x}_A q'$ for some $q' \in Q$, we call $\langle w, x\rangle$ a behavior *of q.*

As an example of the behavior relation, consider the SCA A_{move} in Figure 1. Then the following is true:[12]

$$q_W \xRightarrow{\mathsf{east}\cdot\mathsf{west}\cdot\mathsf{west},\ 5\cdot 5\cdot 0}_{A_{\mathsf{move}}} q_W \qquad\qquad q_E \xRightarrow{\mathsf{west}\cdot\mathsf{stay}_{\mathsf{lon}},\ 5\cdot 5}_{A_{\mathsf{move}}} q_W$$

Thus $\mathsf{east}\cdot\mathsf{west}\cdot\mathsf{west}$ and $\mathsf{west}\cdot\mathsf{stay}_{\mathsf{lon}}$ are behaviors of q_W and q_E respectively. However, $q_W \xRightarrow{\mathsf{west}\cdot\mathsf{west},\ 0\cdot 5}_{A_{\mathsf{move}}} q_W$ does not hold, as $q_W \xrightarrow{\mathsf{west}, 5} q_W$ is not a transition of A_{move}. In essence, the behavior relation allows us to pair the sequences of actions that are allowed in a state with the preferences attached to those actions.

Given a set of behaviors, it is useful to determine which of them can be regarded as optimal. But to speak of optimality, we need to be able to compare sequences of preferences from the underlying c-semiring of the SCA, specifically sequences of preferences that appear on transitions of the SCA. To this end, it is useful to have a handle on the preferences used by an SCA.

Definition 20. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be an SCA. The* preference alphabet *of A, written $\mathfrak{P}(A)$, is defined by $\mathfrak{P}(A) = \{e \in \mathbb{E} : \exists q, q' \in Q, \sigma \in \Sigma.\ q \xrightarrow{\sigma, e} q'\}$. We say that A is* threshold-free *if $t \notin \mathfrak{P}(A)$.*

For technical reasons that become clear later, we often work with threshold-free SCAs.

Because a state may have infinitely many behaviors, there may not be a single behavior among them that is not dominated by another, according to the partial order. For this reason, we offer the option to restrict the set of behaviors under consideration, so as to guarantee the existence of at least one optimal behavior.

Definition 21. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be an SCA and let $\trianglelefteq$ be a partial order on $\mathfrak{P}(A)^*$. Let $L \subseteq \Sigma^*$ and $q \in Q$. Then the* $\trianglelefteq$-optimal behavior within L *of q in A, written $\mathsf{opt}^A_{\trianglelefteq}(q, L)$, is the smallest set satisfying the rule*

$$\frac{q \xRightarrow{w,\ x} q' \qquad w \in L \qquad \forall w' \in L. \left[\text{if } q \xRightarrow{w',\ x'} q'',\ \text{then } x \ntrianglelefteq x'\right]}{\langle w, x\rangle \in \mathsf{opt}^A_{\trianglelefteq}(q, L)}$$

Note that, since $\trianglelefteq$ is a partial order in the above, $\mathsf{opt}^A_{\trianglelefteq}(q, L)$ may contain more than one behavior.

Suppose $\trianglelefteq$ is a partial order, $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ an SCA, q a state of A and $L \subseteq \Sigma$. Inspecting Definition 21, we can see that $\mathsf{opt}^A_{\trianglelefteq}(q, L)$ contains the behaviors of q in L that have a preference word that is not dominated by the preference word of any other behavior of q that is also in L.

[12] Recall that the numeral 0 is not the same as $\mathbf{0}_{\mathbb{W}} = \infty \in \mathbb{W}$.

As an example, suppose $\trianglelefteq$ is the lexicographic order on $\mathbb{W}^*$ induced by $\leq_{\mathbb{W}}$, e.g., $6 \cdot 6 \trianglelefteq 6 \cdot 5 \trianglelefteq 5 \cdot 5 \trianglelefteq 5 \cdot 5 \cdot 5$. If $L = \{\mathsf{west}, \mathsf{north} \cdot \mathsf{stay}_{\mathsf{lon}}, \mathsf{north} \cdot \mathsf{south}\}$ and we consider the SCA A_{diverge} in Figure 2, we can see that

$$\mathsf{opt}_{\trianglelefteq}^{A_{\mathsf{diverge}}}(q_S, L) = \{\langle \mathsf{north} \cdot \mathsf{stay}_{\mathsf{lon}}, 0 \cdot 2\rangle\} \tag{1}$$

This is because west is not an action allowed in q_S (i.e., $q_S \not\xRightarrow{\mathsf{west},\ e} q$ for any $e \in \mathbb{W}$ and q a state of A_{diverge}), and the action word $\mathsf{north} \cdot \mathsf{south}$ is assigned preference word $0 \cdot 5$, which is dominated by the preference word $0 \cdot 2$ of $\mathsf{north} \cdot \mathsf{stay}_{\mathsf{lon}}$. We remark that the lexicographic order, by its nature, orders any prefix of w before w itself; as a result, we find that for $L = \{\mathsf{north} \cdot \mathsf{south}, \mathsf{north}\}$, it holds that

$$\mathsf{opt}_{\trianglelefteq}^{A_{\mathsf{diverge}}}(q_S, L) = \{\langle \mathsf{north} \cdot \mathsf{south}, 0 \cdot 5\rangle\}$$

However, A_{diverge} may conceivably prefer going north to q_M and staying there over going north, and then south again. We investigate partial orders more suitable for this possibility further on in this section.

7.2 Syntax and semantics

We are now ready to formally define the syntax and semantics of PDL_Σ. For brevity, we limit ourselves to formulas in which negation can appear only above the atoms, and the only two modalities are reminiscent of the universal modality in PDL [16]. Moreover, the analog of “programs” in our logic is limited to finite sets of action words, i.e., we do not allow the richness of program expressions as in PDL.

Definition 22. *Let Σ be a CAS, and let At be a finite set of atoms. The set of* valid formulas *for PDL_Σ that uses atoms from At, written $\mathcal{L}_\Sigma^{\mathsf{At}}$, is the smallest set satisfying the rules*

$$\frac{a \in \mathsf{At}}{a \in \mathcal{L}_\Sigma^{\mathsf{At}} \qquad \neg a \in \mathcal{L}_\Sigma^{\mathsf{At}}} \qquad \frac{\phi, \psi \in \mathcal{L}_\Sigma^{\mathsf{At}}}{\phi \vee \psi \in \mathcal{L}_\Sigma^{\mathsf{At}} \qquad \phi \wedge \psi \in \mathcal{L}_\Sigma^{\mathsf{At}}} \qquad \frac{L \subseteq \Sigma \text{ is finite} \qquad \phi \in \mathcal{L}_\Sigma^{\mathsf{At}}}{[L]\,\phi \in \mathcal{L}_\Sigma^{\mathsf{At}} \qquad [L]_*\,\phi \in \mathcal{L}_\Sigma^{\mathsf{At}}}$$

We abbreviate $[\{w_1, w_2, \dots, w_n\}]\,\phi$ with $[w_1, w_2, \dots, w_n]\,\phi$, and likewise for $[\cdot]_*$.

The semantics of a formula of PDL_Σ is given in terms of states of an automaton A and an interpretation of its atoms. We also need a partial order on $\mathfrak{P}(A)^*$, which should order action words in some sensible way (preferably based on the induced order of the c-semiring). The interpretation of (negated) atoms, disjunction and conjunction is done in the familiar ways, but the modalities require a bit more care.

Definition 23. *Let $A = \langle Q, \Sigma, \mathbb{E}, \to, q^0, t\rangle$ be an SCA, At a set of atoms, $\pi : \mathsf{At} \to 2^Q$ a function and $\trianglelefteq$ a partial order on $\mathfrak{P}(A)^*$. The* $\trianglelefteq$-semantics of PDL_Σ for A, *written $\models_{\trianglelefteq}^{A}$, is the smallest relation between Q and $\mathcal{L}_\Sigma^{\mathsf{At}}$ that satisfies the rules*

$$\frac{a \in \mathsf{At} \qquad q \in \pi(a)}{q \models_{\trianglelefteq}^{A} a} \qquad \frac{a \in \mathsf{At} \qquad q \notin \pi(a)}{q \models_{\trianglelefteq}^{A} \neg a} \qquad \frac{q \models_{\trianglelefteq}^{A} \phi \text{ and } q \models_{\trianglelefteq}^{A} \psi}{q \models_{\trianglelefteq}^{A} \phi \wedge \psi} \qquad \frac{q \models_{\trianglelefteq}^{A} \phi \text{ or } q \models_{A} \psi}{q \models_{\trianglelefteq}^{A} \phi \vee \psi}$$

$$\frac{L \subseteq \Sigma \text{ is finite} \qquad \langle q, \phi\rangle \in Q \times \mathcal{L}_\Sigma^{\mathsf{At}} \qquad \forall\, \langle w, x\rangle \in \mathsf{opt}_{\trianglelefteq}^{A}(q, L).\ \left[\text{if } q \xRightarrow{w,\ x}_A q' \text{ then } q' \models_{\trianglelefteq}^{A} \phi\right]}{q \models_{\trianglelefteq}^{A} [L]\,\phi}$$

$$\frac{L \subseteq \Sigma \text{ is finite} \qquad \langle q, \phi\rangle \in Q \times \mathcal{L}_\Sigma^{\mathsf{At}} \qquad \forall n \in \mathbb{N}.\ q \models_{\trianglelefteq}^{A} [L^n]\,\phi}{q \models_{\trianglelefteq}^{A} [L]_*\,\phi}$$

In the sequel, we always specify the function π when it is important for the content of $\models_{\trianglelefteq}^{A}$.

Consider the semantics for the modalities $[\cdot]$ and $[\cdot]_*$ as found in Definition 23. Suppose that A is an SCA, with q a state of A and $\trianglelefteq$ a partial order on $\mathfrak{P}(A)^*$. If $\phi \in \mathcal{L}_\Sigma^{\mathsf{At}}$ and $L \subseteq \Sigma$ is finite, we can see that $q \models_{\trianglelefteq}^{A} [L]\,\phi$ holds for states q in which all $\trianglelefteq$-optimal behaviors of q within L lead to states where ϕ holds. For example, consider the SCA A_{diverge} from Figure 2. As we have seen in Equation 1, $\mathsf{north} \cdot \mathsf{stay}_{\mathsf{lon}}$ is the unique optimal behavior of q_S within $L = \{\mathsf{west}, \mathsf{north} \cdot \mathsf{stay}_{\mathsf{lon}}, \mathsf{north} \cdot \mathsf{south}\}$, which leads to q_M. Thus, if CENTER is an atom such that $\pi(\text{CENTER}) = \{q_M\}$, then $q_S \models_{\trianglelefteq}^{A} [L]$ CENTER holds, since all $\trianglelefteq$-optimal behaviors of q_S in L lead to states where CENTER holds.

The semantics of the modality $[\cdot]_*$ is slightly more complex. From Definition 23, we can see that $q \models_{\trianglelefteq}^{A} [L]_*\,\phi$ holds if $q \models_{\trianglelefteq}^{A} [L^n]\,\phi$ holds for all $n \in \mathbb{N}$. We can thus surmise that, if we consider any $\trianglelefteq$-optimal behavior q within L^n, then this behavior leads to a state where ϕ holds. More informally, $q \models_{\trianglelefteq}^{A} [L]_*\,\phi$ holds when, given that we need to execute n action words from L, we can only end up in a state where ϕ holds if we choose an $\trianglelefteq$-optimal behavior within L^n. We note that, as the preference of an action may differ based on the state of the

SCA, it is not the case that we can just concatenate n optimal behaviors from L to obtain an optimal behavior in L^n.

As an example of the use of $[\cdot]_*$, consider the SCA $A_{\mathsf{move}} \triangleright A_{\mathsf{diverge}}$ as sketched in Figure 3. If we want to assert that the optimal behavior of the agent in $q_{W,M}$, when restricted to move longitudinally after every two moves, is such that the agent remains centered after every six steps, we can set

$$L = \mathcal{M}^2 \cdot \{\mathsf{south}, \mathsf{southeast}, \mathsf{north}, \mathsf{northeast}\}$$

and assert that $q_{W,M} \models^A_\trianglelefteq [L^2]_*$ CENTER holds.

We note that for the fragment of PDL$_\Sigma$ that does not include the modality $[\cdot]_*$, one can compute $\mathsf{opt}^A_\trianglelefteq(q, L)$ for any SCA $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ with $q \in Q$ and $L \subseteq \Sigma$ finite, and a given (decidable) partial order $\trianglelefteq$; obviously, we can then decide $q \models^A_\trianglelefteq \phi$ by means of a simple recursive procedure. To decide full PDL$_\Sigma$, we need to use the particulars of $\trianglelefteq$.

7.3 Partial orders for idling

We now propose two partial orders that can be used as the partial order in the $\trianglelefteq$-semantics as outlined above. Both of these partial orders are based on the idea that, for an agent, it is sometimes better to *not* perform any action, i.e., to *idle*. Recall that, in the case for A_{diverge} discussed in the above, $\mathsf{north} \cdot \mathsf{south}$ was preferred over north in q_S by the lexicographic ordering on $\mathbb{W}^*$. An ordering that incorporates idling can help, by ordering north over $\mathsf{north} \cdot \mathsf{south}$ in q_S, based on the knowledge that it is better to “go north to the center of the track and then do nothing”, than “go north and to the center and then go south again”. This is modeled by assuming the existence of an idling action that is available in all states and does not affect the state of the agent in any meaningful way. Furthermore, we assume that the preference for this action is also fixed.

The orders we discuss can be derived from any threshold-free SCA A. To be precise, given a threshold-free SCA A with threshold value t, we derive a partial order on $\mathfrak{P}(A)^*$ such that t is the preference of idling (note that our interpretation of t differs here from that in Section 6).

Given a preference word $w \in \mathfrak{P}(A)^*$, we need to construct the *idling set* of of w, as outlined below.

Definition 24. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be an SCA. The* threshold projection of *A is the function $\mathfrak{p}_A : (\mathfrak{P}(A) \cup \{t\})^* \rightarrow \mathfrak{P}(A)^*$ that deletes all occurrences of t from a word w. Let $w \in \mathfrak{P}(A)^*$ and $n \in \mathbb{N}$; the* n-augmented idling set of *w, denoted by $\mathsf{idle}_A(w, n)$, is defined by*

$$\mathsf{idle}_A(w, n) = \mathfrak{p}_A^{-1}(\mathfrak{p}_A(w)) \cap \mathbb{E}^n$$

For instance, let A be an SCA with threshold value t and $e \in \mathfrak{P}(A)$ such that $e \neq t$; then $\mathfrak{p}_A(etete) = eee$. Also:

$$\mathsf{idle}_A(ee, 4) = \{eett, etet, ette, tete, ttee\}$$

Note that, when $w \in \mathfrak{p}_A(\mathfrak{P}(A)^*)$, it is sufficient to show that $|w'| = n$ and $\mathfrak{p}_A(w') = w$ for $w' \in \mathsf{idle}_A(w, n)$ to hold; this condition is immediately satisfied by action words of a threshold-free SCA.

Generalized pointwise order We are now ready to define our first proposed partial order that incorporates idling.

Definition 25. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA. The operator $\bigvee_{\mathbb{E}^n} : 2^{\mathbb{E}^n} \rightarrow \mathbb{E}^n$ is defined for $n \in \mathbb{N}$ and $E \subseteq \mathbb{E}^n$ as follows:*

$$\bigvee_{\mathbb{E}^n} E = \bigvee_{\mathbb{E}} \mathsf{Pr}_1(E) \cdot \bigvee_{\mathbb{E}} \mathsf{Pr}_2(E) \cdot \cdots \cdot \bigvee_{\mathbb{E}} \mathsf{Pr}_n(E)$$

The operator $\bigvee_{\mathbb{E}^n}$ induces a partial order $\leq_{\mathbb{E}^n}$ on $\mathbb{E}^n$ in the same manner as seen before: if $w, x \in \mathbb{E}^n$ such that $\bigvee_{\mathbb{E}^n}\{w, x\} = x$, then $w \leq_{\mathbb{E}^n} x$. The generalized pointwise order *of A, denoted $\preceq_A$, is the smallest relation on $\mathfrak{P}(A)^*$ that satisfies the rule*

$$\frac{w, x \in \mathfrak{P}(A)^* \qquad p = \max(|w|, |x|) \qquad \bigvee_{\mathbb{E}^p} \mathsf{idle}(w, p) \leq_{\mathbb{E}^p} \bigwedge_{\mathbb{E}^p} \mathsf{idle}(x, p)}{w \preceq_A x}$$

To get a feeling for the relation $\preceq_A$, we reconsider A_{move} (Figure 1). In state q_W, the action word $\mathsf{west} \cdot \mathsf{east}$ has preference word $5 \cdot 5$, while the action word west has preference word 5. If we suppose that the threshold value t is 1 (note that this makes A_{move} threshold-free), then $\mathsf{idle}_{A_{\mathsf{move}}}(5 \cdot 5, 2) = \{5 \cdot 5\}$ and $\mathsf{idle}_{A_{\mathsf{move}}}(5, 2) = \{5 \cdot 1, 1 \cdot 5\}$. Accordingly, we find that

$$\bigvee_{\mathbb{W}^2}\{5 \cdot 5\} = 5 \cdot 5 \leq_{\mathbb{W}^2} 5 \cdot 5 = \bigwedge_{\mathbb{W}^2}\{5 \cdot 1, 1 \cdot 5\}$$

$$\bigvee_{\mathbb{W}^2}\{5 \cdot 1, 1 \cdot 5\} = 1 \cdot 1 \not\leq_{\mathbb{W}^2} 5 \cdot 5 = \bigwedge_{\mathbb{W}^2}\{5 \cdot 5\}$$

and therefore that $5 \not\preceq_A 5 \cdot 5$, but $5 \cdot 5 \preceq_A 5$, i.e., the preference word of the action word west is considered better than that of the action word $\mathsf{west} \cdot \mathsf{east}$, signaling that we prefer the agent not to keep switching directions.

To see that $\preceq_A$ is indeed a partial order, we need an alternative characterization of the application of $\bigwedge_{\mathbb{E}^n}$ and $\bigvee_{\mathbb{E}^n}$ to idling sets, based on the following functions.

Definition 26. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA. We define the functions $\sharp_A : \mathbb{E}^* \to \mathbb{E}^*$ and $\flat_A : \mathbb{E}^* \to \mathbb{E}^*$ inductively, as follows:*

$$\sharp_A(w) = \begin{cases} t & w = \epsilon \\ \sharp_A(w') \cdot e \vee_{\mathbb{E}^{|w|+1}} w' \cdot e \cdot t & w = w' \cdot e \end{cases} \qquad \flat_A(w) = \begin{cases} t & w = \epsilon \\ \flat_A(w') \cdot e \wedge_{\mathbb{E}^{|w|+1}} w' \cdot e \cdot t & w = w' \cdot e \end{cases}$$

Using these functions, we can then establish the following (proofs appear in Appendix B).

Lemma 19. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA. If $w, x \in \mathfrak{P}(A)^*$ such that $|w| = n = |x|$ and $w \leq_{\mathbb{E}^n} x$, then we have that*

$$\sharp_A(w) \leq_{\mathbb{E}^{n+1}} \sharp_A(x)$$
$$\flat_A(x) \leq_{\mathbb{E}^{n+1}} \flat_A(w)$$

Lemma 20. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA. Then for $w \in \mathbb{E}^*$ and $n \in \mathbb{N}$ with $|w| \leq n$, the following equalities hold:*

$$\bigvee \mathsf{idle}_A(w, n) = \sharp_A^{n-|w|}(w)$$
$$\bigwedge \mathsf{idle}_A(w, n) = \flat_A^{n-|w|}(w)$$

It is useful to note that for an SCA A with $w \in \mathfrak{P}(A)^*$, we can compute $\sharp_A(w)$ (respectively $\flat_A(w)$) reasonably easy: by inspecting Definition 26, we find that we need approximately $|w|^2$ applications of $\vee$ (respectively $\wedge$) to compute $\sharp_A(w)$ (respectively $\flat_A(w)$). Lemma 20 then provides us with a computationally feasible way to establish whether for an SCA A with $w, x \in \mathfrak{P}(A)^*$ it holds that $w \preceq_A x$: simply apply $\sharp_A$ or $\flat_A$ as necessary to obtain action words of equal length, and compare the results. To decide whether $w \preceq_A x$, we thus need on the order of $\max(|w|, |x|)^3$ applications of $\vee$ or $\wedge$ in the worst case.

Using these lemmas, one can now establish that $\preceq_A$ is indeed a partial order, as long as A is threshold-free.

Theorem 1. *Let A be an SCA. If A is threshold-free, then $\preceq_A$ is a partial order on $\mathfrak{P}(A)^*$.*

Proof. Throughout the following, let $w, x \in \mathfrak{P}(A)^*$ and $p = \max(|w|, |x|)$.

If $w \in \mathfrak{P}(A)^*$, then $\mathfrak{p}_A(w) = w$ (as A is threshold-free, so w does not contain any occurrence of t); therefore, $w \in \mathsf{idle}_A(w, |w|)$. Moreover, if $w' \in \mathsf{idle}_A(w, |w|)$, then $|w| = |w'|$ and $\mathfrak{p}_A(w') = w$; since $|\mathfrak{p}_A(w')| = |w|$, we know that w' does not contain any occurrence of t, and therefore necessarily $w' = w$. Consequently, $\mathsf{idle}_A(w, |w|) = \{w\}$.

For reflexivity, assume that $w = x$ and observe that $w \preceq_A x$ holds, since we can derive

$$\begin{aligned} \bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(w, p) &= \bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(w, |w|) \\ &= w \leq_{\mathbb{E}^p} x \\ &= \bigwedge\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, |x|) \\ &= \bigwedge\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, p) \end{aligned}$$

For antisymmetry, consider that when $w \preceq_A x$ and $x \preceq_A w$, it holds that

$$\bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(w, p) \leq_{\mathbb{E}^p} \bigwedge\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, p) \leq_{\mathbb{E}^p} \bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, p) \leq_{\mathbb{E}^p} \bigwedge\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(w, p)$$

Assuming (without loss of generality) that $p = |w|$, we can conclude that

$$w \leq_{\mathbb{E}^p} \bigwedge\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, |w|) \leq_{\mathbb{E}^p} \bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, |w|) \leq_{\mathbb{E}^p} w$$

Consequently, for all elements $x' \in \mathsf{idle}_A(x, |w|)$ it holds that $w \leq_{\mathbb{E}^p} x' \leq_{\mathbb{E}^p} w$, and therefore by antisymmetry of $\leq_{\mathbb{E}^p}$ we have that $w = x'$. Thus, $\mathsf{idle}_A(x, |w|) = \{w\}$ (note that $\mathsf{idle}_A(x, |w|)$ is not empty, because $|x| \leq |w|$). From this we know that $\mathfrak{p}_A(w) = x$. But since $w \in \mathfrak{P}(A)^*$ and A is threshold-free, it follows that $w = \mathfrak{p}_A(w) = x$.

For transitivity, let $w \preceq_A x$ and $x \preceq_A y$. We now need to show that $w \preceq_A y$; this is done by a case distinction on the order of the lengths. Let $q = \max(|x|, |y|)$ and $r = \max(|w|, |x|, |y|)$. From $w \preceq_A x$ and $x \preceq_A y$, we have

$$\bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(w, p) \leq_{\mathbb{E}^p} \bigwedge\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, p)$$
$$\bigvee\nolimits_{\mathbb{E}^q} \mathsf{idle}_A(x, q) \leq_{\mathbb{E}^q} \bigwedge\nolimits_{\mathbb{E}^q} \mathsf{idle}_A(y, q)$$

(i) For the first case, let $|w|, |x| \leq |y|$. Then $r = q$ and therefore we can derive

$$\begin{aligned}
\bigvee\nolimits_{\mathbb{E}^r} \mathsf{idle}_A(w, r) &= \sharp_A^{r-|w|}(w) && \text{(Lemma 20)} \\
&= \sharp_A^{r-p}(\sharp_A^{p-|w|}(w)) \\
&= \sharp_A^{r-p}\left(\bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(w, p)\right) && \text{(Lemma 20)} \\
&\leq_{\mathbb{E}^n} \sharp_A^{r-p}\left(\bigwedge\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, p)\right) && \text{(Lemma 19)} \\
&\leq_{\mathbb{E}^n} \sharp_A^{r-p}\left(\bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, p)\right) && \text{(Lemma 19)} \\
&= \sharp_A^{r-p}(\sharp_A^{p-|x|}(x)) && \text{(Lemma 20)} \\
&= \sharp_A^{r-|x|}(x) \\
&= \bigvee\nolimits_{\mathbb{E}^r} \mathsf{idle}_A(x, r) && \text{(Lemma 20)} \\
&\leq_{\mathbb{E}^n} \bigwedge\nolimits_{\mathbb{E}^r} \mathsf{idle}_A(y, r)
\end{aligned}$$

Thus $w \preceq_A y$.

(ii) For the second case, assume that $|w|, |y| \leq |x|$. Then $r = p = q$ and we can derive

$$\begin{aligned}
\bigvee\nolimits_{\mathbb{E}^r} \mathsf{idle}_A(w, r) &= \bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(w, p) \\
&\leq_{\mathbb{E}^p} \bigwedge\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, p) \\
&\leq_{\mathbb{E}^p} \bigvee\nolimits_{\mathbb{E}^p} \mathsf{idle}_A(x, p) \\
&= \bigvee\nolimits_{\mathbb{E}^q} \mathsf{idle}_A(x, q) \\
&\leq_{\mathbb{E}^q} \bigwedge\nolimits_{\mathbb{E}^q} \mathsf{idle}_A(y, q) \\
&= \bigwedge\nolimits_{\mathbb{E}^r} \mathsf{idle}_A(y, r)
\end{aligned}$$

Thus $w \preceq_A y$.

(iii) The third case, where $|x|, |y| \leq |w|$ is completely analogous to the first case, except that the proof uses the second parts of Lemma 20 and Lemma 19. We thus derive that $w \preceq_A x$ in this case, too. □

We can see that the proof of Theorem 1 depends on the SCA being threshold free for reflexivity, antisymmetry and transitivity: for reflexivity, we use the threshold-freeness of A to show that $\mathsf{idle}(w, |w|) = \{w\}$, for antisymmetry we use that $\mathfrak{p}_A(w) = w$ and for transitivity we use Lemma 20, which also requires A to be threshold-free.

Generalized lexicographic order The generalized pointwise order defined above makes no distinction between the positions of a preference word. As a result, it can easily be seen that it is not a total order for most SCAs; for example, for A_{diverge}, where the underlying c-semiring is $\mathbb{W}$, we see that $0 \cdot 5 \in \mathbb{W}^*$ is not related to $5 \cdot 4 \cdot 0 \in \mathbb{W}^*$ by $\preceq_{A_{\mathsf{diverge}}}$ if we assume that the threshold value is 3. We now propose a relation on preference words that does include such a distinction between positions.

Before we talk about the generalized lexicographic order, we recall from Section 3 that the partial order $\leq_{\mathbb{E}}$ on $\mathbb{E}$ induces a lexicographic partial order $\leq_{\mathbb{E}^*}$ on $\mathbb{E}^*$. Using the lexicographic order on $\mathbb{E}^*$, we can then define the generalized lexicographic order on the preference alphabet of an SCA, as follows.

Definition 27. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be an SCA, and let $n \in \mathbb{N}$. We write $\sqsubseteq_A^n$ and $\sqsubseteq_A$ for the smallest relations on $\{w \in \mathfrak{P}(A)^* : |w| \leq n\}$, and $\mathfrak{P}(A)^*$ respectively satisfying the rules*

$$\frac{\forall w' \in \mathsf{idle}_A(w, n).\ \exists x' \in \mathsf{idle}_A(x, n).\ w' \leq_{\mathbb{E}^*} x'}{w \sqsubseteq_A^n x} \qquad \frac{w, x \in \mathfrak{P}(A)^* \qquad n = \max(|w|, |x|) \qquad w \sqsubseteq_A^n x}{w \sqsubseteq_A x}$$

To get a feeling for the generalized lexicographic order, consider the action words $\mathsf{west} \cdot \mathsf{stay}_{\mathsf{lon}}$ and $\mathsf{east} \cdot \mathsf{east} \cdot \mathsf{east}$, which represent sequences of actions available in state q_W of A_{move} (in Figure 1). The preference words attached to these are $0 \cdot 5$ and $5 \cdot 0 \cdot 0$ respectively. If we set the threshold value of A_{move} to $t = 3$, we find that $5 \cdot 0 \cdot 0 \leq_{\mathbb{E}^*} 0 \cdot 5 \cdot 3$, thus $5 \cdot 0 \cdot 0 \sqsubseteq_{A_{\mathsf{move}}}^3 0 \cdot 5$ and consequently $5 \cdot 0 \cdot 0 \sqsubseteq_{A_{\mathsf{move}}} 0 \cdot 5$. Therefore, the action word $\mathsf{west} \cdot \mathsf{stay}_{\mathsf{lon}}$ is preferred over the action word $\mathsf{east} \cdot \mathsf{east} \cdot \mathsf{east}$.

Informally, the generalized lexicographic order encodes that preferences for actions closer to the present are more important than preferences for actions further in the future. Also, a shorter action word may be preferred over a longer action word in cases where, by inserting the idling preference, it can achieve a better preference. Vice versa, a longer action word is preferred over a shorter action word if the shorter action word cannot possibly achieve the preference of the longer action word by inserting idling.

Similar to the generalized pointwise order, we can show that $\sqsubseteq_A$ is a partial order if A is a threshold-free SCA. We first make the following observations about $\sqsubseteq_A^n$; the proofs are fairly easy and appear in Appendix B.

Lemma 21. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA and let $n \in \mathbb{N}$. The relation $\sqsubseteq_A^n$ is reflexive.*

Lemma 22. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA, and let $n \in \mathbb{N}$. If $w, x \in \mathfrak{P}(A)^*$ are such that $n = \max(|w|, |x|)$ and $w \sqsubseteq_A^n x \sqsubseteq_A^n w$, then $w = x$.*

Lemma 23. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA, and let $n \in \mathbb{N}$. The relation $\sqsubseteq_A^n$ is transitive.*

The following lemma is slightly harder to prove, but very important to show that $\sqsubseteq_A$ is a partial order: it allows us to vary n in $\sqsubseteq_A^n$, as long as we keep comparing words of at most length n. Again, a proof is provided in Appendix B.

Lemma 24. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA. If $w, x \in \mathfrak{P}(A)^*$ and $n \in \mathbb{N}$ such that $|w|, |x| \leq n$, then $w \sqsubseteq_A^n x$ if and only if $w \sqsubseteq_A^{n+1} x$.*

With these lemma's in place, we are now ready to prove that $\sqsubseteq_A$ is indeed a partial order for threshold-free SCAs.

Theorem 2. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA. The relation $\sqsubseteq_A$ is a partial order.*

Proof. Let $w \in \mathfrak{P}(A)^*$. Then $w \sqsubseteq_A w$ since $w \sqsubseteq_A^{|w|} w$, the latter being a consequence of Lemma 21. It follows that $\sqsubseteq$ is reflexive. Likewise, antisymmetry of $\sqsubseteq_A$ is a consequence of Lemma 22; let $w, x \in \mathfrak{P}(A)^*$ and set $p = \max(|w|, |x|)$. Then $w \sqsubseteq_A x \sqsubseteq_A w$ is due to $w \sqsubseteq_A^p x \sqsubseteq_A^p w$. By Lemma 22 we can conclude that $w = x$.

A bit more work is required to show transitivity. Let $w, x, y \in h(E^*)$ be such that $w \sqsubseteq x \sqsubseteq y$. Then choose the constants p, q, r, s as follows:

$$p = \max(|w|, |x|) \qquad q = \max(|x|, |y|) \qquad r = \max(|w|, |y|) \qquad s = \max(|w|, |x|, |y|)$$

Now we know that $w \sqsubseteq_A^p x$ and $x \sqsubseteq_A^q y$. By Lemma 24, we know that $w \sqsubseteq_A^s x \sqsubseteq_A^s y$ and by Lemma 23 we can derive that $w \sqsubseteq_A^s y$. Again by Lemma 24, we can then derive that $w \sqsubseteq_A^r y$ and therefore conclude $w \sqsubseteq_A y$. □

Finally, the following observation is also useful

Lemma 25. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA, and let $n \in \mathbb{N}$. If $\leq_{\mathbb{E}}$ is a total order, so is $\sqsubseteq_A$.*

Proof. First, note that if $\leq_{\mathbb{E}}$ is a total order, then so is $\leq_{\mathbb{E}^*}$. In particular, this implies that for $w, x \in \mathfrak{P}(A)^*$, whenever $w \not\leq_{\mathbb{E}^*} x$ we have $x \leq_{\mathbb{E}^*} w$.

In Theorem 2, we have shown that $\sqsubseteq_A$ is a partial order. Totality remains to be shown. To show that $\sqsubseteq_A$ is total, we need to show that for all $w, x \in \mathfrak{P}(A)^*$ with $n = \max(|w|, |x|)$ we have either $w \sqsubseteq_A^n x$ or $x \sqsubseteq_A^n w$. We already know that either $w \sqsubseteq_A^n x$ or $w \not\sqsubseteq_A^n x$ holds. In the former case, we are done. In the latter case we can derive the following:

$$\begin{aligned}
w \not\sqsubseteq_A^n x \text{ then } & \exists w' \in \mathsf{idle}_A(w, n).\ \forall x' \in \mathsf{idle}_A(x, n).\ w' \not\leq_{\mathbb{E}^*} x' && \text{(Definition 27)} \\
\text{then } & \exists w' \in \mathsf{idle}_A(w, n).\ \forall x' \in \mathsf{idle}_A(x, n).\ x' \leq_{\mathbb{E}^*} w' && \text{(totality of } \leq_{\mathbb{E}^*}\text{)} \\
\text{then } & \forall x' \in \mathsf{idle}_A(x, n).\ \exists w' \in \mathsf{idle}_A(w, n).\ x' \leq_{\mathbb{E}^*} w' && \text{(first-order reasoning)} \\
\text{then } & x \sqsubseteq_A^n w && \text{(Definition 27)}
\end{aligned}$$

We have thus shown that either $w \sqsubseteq_A^n x$ or $x \sqsubseteq_A^n w$, proving that $\sqsubseteq_A$ is total. □

A note on compositionality As seen in the above, $\preceq_A$ and $\sqsubseteq_A$ are partial orders only when A is a threshold-free SCA. This is somewhat of a flaw with regard to our objective of compositionality, in that threshold-freeness of SCAs is not a compositional property. It is fairly easy to come up with SCAs A_1 and A_2 with underlying c-semiring $\mathbb{E}$ that are threshold-free, while $A_1 \bowtie A_2$ is not; essentially all one needs to do is find distinct e_1, e_2, e_3, e_4 such that $e_1 \otimes e_2 = e_3 \otimes e_4$.

7.4 Towards model checking

Suppose that A is an SCA, we can decide whether $q \models^A_\trianglelefteq \phi$ holds, and that the set

$$R = \bigcup \{\mathsf{opt}^\trianglelefteq_A(q, L^n) : n \in \mathbb{N}\} \subseteq \Sigma^* \times \mathfrak{P}(A)^*$$

has a regular structure. Then we can sketch the following decision procedure for $q \models^A_\trianglelefteq [L]_* \, \phi$:

1. Reinterpret A as a finite automaton A_F, with q as initial state, the set $\Sigma \times \mathfrak{P}(A)$[13] as alphabet and the states $q' \in Q$ such that $q' \models^A_\trianglelefteq \phi$ as accepting states.
2. Construct a finite automaton A_R that accepts R.
3. Decide whether $L(A_R) \subseteq L(A_F)$ holds; if this is the case, then any $\langle w, x\rangle \in \mathsf{opt}^\trianglelefteq_A(q, L^n)$ induces a path in A that leads to a state where ϕ holds.

It seems not altogether unreasonable to expect that there are cases where R is regular. Consider, for example, the SCA depicted in Figure 6, which uses the c-semiring $\mathbb{W}$ and the CAS $\mathcal{M}$.

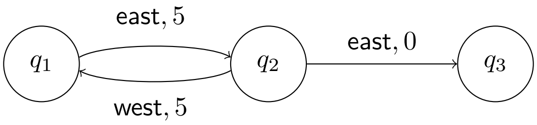


Figure 6: An SCA with a regular structure on maximally preferred actions.

If we set $L = \{\mathsf{west}, \mathsf{east}\}$ and assume the threshold value to be $t = 10$ then one can infer from the structure of A that the following is true:

$$\bigcup \{\mathsf{opt}^A_{\sqsubseteq_A}(q_2, L^n) : n \in \mathbb{N}\} = \left\{\left\langle 5^{2n} \cdot 0, (\mathsf{west} \cdot \mathsf{east})^n \cdot \mathsf{east}\right\rangle : n \in \mathbb{N}\right\} \cup \left\{\left\langle 5^{2n}, (\mathsf{west} \cdot \mathsf{east})^n\right\rangle : n \in \mathbb{N}\right\}$$

The set above is regular, but it is not quite clear under which circumstances such a pattern arises, and how to construct one when it does.

We found that if one concentrates purely on preference words, i.e., when one ignores the indirection added by pairing of action words to preference words, it is possible to find a regular structure, as outlined in the theorems below. First, we need to define what it means for a preference word to be *effective*; intuitively, effective preference words are words that consist of preference values that are not dominated by the idling preference of the SCA.

Definition 28. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be an SCA. We call a word $w \in \mathfrak{P}(A)^*$* effective *when $w = \epsilon$, or when $w = e \cdot w'$ with $e <_\mathbb{E} t$ and w' is effective. We call w non-effective when it is not effective.*

We are now ready to state the main results of this section. Their proofs are reasonably complicated, and require a series of lemmas about the generalized lexicographic order; we refer to Appendix C for a full treatment.

Theorem 3. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA. Let $L \subseteq \mathfrak{P}(A)^*$ be non-empty and finite, and let $\leq_{\mathbb{E}^*}$ be total. Let $w = \max_{\sqsubseteq_A}(L)$ and choose the longest $x \in \mathsf{prefix}(w) \cap L$ such that $x \cdot w$ is the $\leq_{\mathbb{E}^*}$-maximum of $(\mathsf{prefix}(w) \cap L) \cdot w$.*[14] *If w is effective or x is a proper prefix of w, then $\max_{\sqsubseteq_A}(L^n) = x^{n-1} \cdot w$ for $n \geq 1$.*

Theorem 4. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA. Let $L \subseteq \mathfrak{P}(A)^*$ be non-empty and finite, and let $\leq_{\mathbb{E}^*}$ be total. Let $w = \max_{\sqsubseteq_A}(L)$. If w is non-effective and $w \cdot w$ is the $\leq_{\mathbb{E}^*}$-maximum of $(\mathsf{prefix}(w) \cap L) \cdot w$, then $\max_{\sqsubseteq_A}(L^n) = w \cdot z^{n-1}$ for $n \geq 1$, where z is the $\leq_{\mathbb{E}^*}$-maximum of the shortest elements in L.*

Corollary 1. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA. Let $L \subseteq \mathfrak{P}(A)^*$ be non-empty and finite, and let $\leq_{\mathbb{E}^*}$ be total. Then $\{\max_{\sqsubseteq_A}(L^n) : n \in \mathbb{N}\}$ is regular.*

Proof. Follows from Theorem 3 and Theorem 4, when one observes that the conditions posed on w cover all possible cases, and that either case makes the claimed set a regular language. □

Our intuition is that the theorems above are a useful first step towards investigating the conditions for regularity of the set R, and therefore towards model checking for (a fragment of) PDL$_\Sigma$. Unfortunately, we were unable to come up with a way to lift the regularity of the set described above to a regular description of R. The problem here is exactly the indirection: if $\langle w, x\rangle$ is a behavior of q_1 in A, and $\langle w, x'\rangle$ is a behavior of q_2 in A, then it is very possible that $x \neq x'$, and thus that $\langle w, x\rangle$ is not optimal behavior in q_1, while $\langle w, x'\rangle$ is optimal behavior in q_2.

[13] Here, we idenitfy the word $\langle \sigma_1, e_1\rangle \langle \sigma_2, e_2\rangle \cdots \langle \sigma_n, e_n\rangle \in (\Sigma \times \mathfrak{P}(A))^*$ with $\langle \sigma_1\sigma_2 \cdots \sigma_n, e_1e_2 \cdots e_n\rangle \in \Sigma^* \times \mathfrak{P}(A)^*$.

[14] Such an x always exists, for $w \cdot w \in (\mathsf{prefix}(w) \cap L) \cdot w$ and $(\mathsf{prefix}(w) \cap L) \cdot w$ is finite.

8 Conclusion

We have proposed a framework for modeling agents using an automata formalism called Soft Component Automata (SCAs). Since actions and their preferences originate from well-defined algebraic structures (Component Action Systems and Constraint Semirings, respectively), we obtain easily definable composition operators. These operators compose SCAs such that, in the composition, composed actions meaningfully represent their component actions, and practically useful preferences for composed actions arise from the preferences of the component actions. Using SCAs and composition, one can specify the preferences and actions available in each state of an agent concisely.

We then considered two approaches to verification of SCAs. For the first approach, we reduced SCAs to Büchi-automata and proposed a logic based on LTL that reflected their compositional nature. We showed that model checking of SCAs using this paradigm is feasible, and sketched a decision procedure for the logic based on a well-known decision procedure for LTL. We furthermore argued in favor of using LTL for SCAs, by showing that one can trace undesired behavior back to the component (or combination of components) it originated from.

An alternative approach that we consider for verification for SCAs is based on PDL. Here, our formulas make assertions about the optimal behavior of the SCA when restricted to a set of allowed actions, for a user-provided partial order that dictates which behavior is optimal. We provided two such partial orders based on the idea that an agent can elect not to perform any action, and that doing so may sometimes be preferable over performing an action. We then briefly expanded upon a possible way to obtain a model checking procedure for our logic based on regularity of optimal behavior, and argued that for at least one instantiation of the partial order, there are indications that such a regular structure exists under some circumstances.

9 Further work

Further work in the theory of c-semirings may have its application in the theory of SCAs. For example, given two c-semirings that model separate concerns, it may be possible to construct a c-semiring that reflects that we want to satisfy both concerns, but in which we prefer to keep both concerns at least somewhat satisfied, i.e., a preference value where one concern has the maximum preference and the other has a very low preference should not be preferred over a preference value where both concerns are reasonably high. Such a “utilitarian” composition can then give rise to a similar composition operator for SCAs.

In [20], Koehler and Clarke showed that all Port Automata can be constructed from a small set of atomic Port Automata. Since Port Automata can be viewed as an instance of SCAs for a particular CAS, it would be interesting to see if their techniques can be generalized, to the point where we can obtain a sufficient condition on a CAS that ensures that a similar decomposition result holds for SCAs.

The desire for a computationally more feasible construction of Büchi-automata from formulas in LTL_Σ is also a good starting point for further work. If such a construction exist, it would mean that we can indeed apply the decision procedure proposed, as well as the methods for diagnostics of undesired behavior. Furthermore, in [2], Baier et al. propose an extension of LTL, called LTL_{IO}, that incorporates regular expressions that generalize the U-operator to the point where it can capture the pathway modalities of $[\cdot]$ and $\langle\cdot\rangle$ found in PDL; lifting this extension to LTL_Σ seems like a useful extension, particularly if we can generalize the regular expressions using the structure imposed by the CAS.

A serious flaw in our proposal for PDL_Σ is that it does not incorporate composition. Further research is required to see if composition can given a place in PDL_Σ; perhaps one can look into generalizing the intersection operator found in some extensions of classic PDL to work on the level of a CAS. Compositionality is also lacking in PDL_Σ where threshold-free automata are concerned. More investigation of the partial orders is necessary to see if this problem can be resolved.

Lastly, we leave open the question of whether an efficient model checking procedure for PDL_Σ can exist for particular instances of the partial order. Further investigation of the generalized lexicographic order is necessary to see if our observations of regularity can be lifted to the set of (restricted) optimal behaviors of a state. Similar investigations of the generalized pointwise order can also prove useful in this regard.

A Proofs for Subsection 3.2

Lemma 26. *Let $\mathbb{E}$ be a c-semiring with $e \in \mathbb{E}$ and $e \neq \mathbf{0}_{\mathbb{E}}$, and let $\mathbf{t}_e : \mathbb{E} \to \mathbb{B}$ be the function defined by*

$$\mathbf{t}_e(e') = \begin{cases} \top & e \leq_{\mathbb{E}} e' \\ \bot & \textit{otherwise} \end{cases}$$

Then $\mathbf{t}_e$ is e-reflecting. Furthermore, if $\leq_{\mathbb{E}}$ is total, $\otimes_{\mathbb{E}}$ is idempotent and $e \neq \mathbf{0}_{\mathbb{E}}$, then $\mathbf{t}_e$ is a homomorphism.

Proof. We first show that $\mathbf{t}_e$ is t-reflecting. Let $e' \in \mathbb{E}$. Observe that $\mathbf{t}_e(e) = \top$. If $e \leq e'$, then $\mathbf{t}_e(e') = \top$, thus $\mathbf{t}_e(e) \leq \mathbf{t}_e(e')$; if on the other hand $e \not\leq e'$, then $\mathbf{t}_e(e') = \bot$, thus $\mathbf{t}_e(e) \not\leq \mathbf{t}_e(e')$.

Now assume that $\leq_{\mathbb{E}}$ is a total order, $\otimes_{\mathbb{E}}$ is idempotent and $e \neq \mathbf{0}_{\mathbb{E}}$. We show that $\mathbf{t}_e$ is a homomorphism in this case. Note that since $e \neq \mathbf{0}_{\mathbb{E}}$, it holds that $e \not\leq_{\mathbb{E}} \mathbf{0}_{\mathbb{E}}$, thus $\mathbf{t}_e(\mathbf{0}_{\mathbb{E}}) = \bot = \mathbf{0}_{\mathbb{B}}$. Also, $\mathbf{t}_e(\mathbf{1}_{\mathbb{E}}) = \top = \mathbf{1}_{\mathbb{B}}$.

Let $E \subseteq \mathbb{E}$. To see that $\mathbf{t}_e\left(\bigvee_{\mathbb{E}} E\right) = \bigvee_{\mathbb{B}} \mathbf{t}_e(E)$, consider two cases. If there exists an $e' \in E$ such that $e \leq_{\mathbb{E}} e'$, then since $e' \leq \bigvee_{\mathbb{E}} E$ we have that $e \leq \bigvee_{\mathbb{E}} E$. Consequently, $\mathbf{t}_t\left(\bigvee_{\mathbb{E}} E\right) = \top$, and $\top \in \mathbf{t}_e(E)$, thus $\bigvee_{\mathbb{B}} \mathbf{t}_e(E) = \top$. If on the other hand, $e \not\leq e'$ for all $e' \in E$, we know that $e' < e$ for all $e' \in E$. But then $\bigvee_{\mathbb{E}} E < e$, thus $e \not\leq \bigvee_{\mathbb{E}} E$. Consequently, $\mathbf{t}_e(\bigvee_{\mathbb{E}} E) = \bot$. Also, $\mathbf{t}_e(E) = \{\bot\}$, thus $\bigvee_{\mathbb{E}} \mathbf{t}_e(E) = \bot$. In either case, we find that $\mathbf{t}_e\left(\bigvee_{\mathbb{E}} E\right) = \bigvee_{\mathbb{B}} \mathbf{t}_e(E)$.

To see that $\mathbf{t}_e(e' \otimes_{\mathbb{E}} e'') = \mathbf{t}_e(e') \otimes_{\mathbb{B}} \mathbf{t}_e(e'')$, we also consider two cases. If $e' < e$ or $e'' < e$, we have that $e' \otimes_{\mathbb{E}} e'' < e$ by intensivity (Lemma 3). Thus $\mathbf{t}_e(e' \otimes_{\mathbb{E}} e'') = \bot = \bot \otimes_{\mathbb{B}} \bot = \mathbf{t}_e(e') \otimes_{\mathbb{B}} \mathbf{t}_e(e'')$. If on the other hand $e \leq e', e''$, we have that $e = e \otimes_{\mathbb{E}} e \leq e' \otimes_{\mathbb{E}} e''$, by idempotency and monotonicity of $\otimes_{\mathbb{E}}$ (Lemma 2). Accordingly, $\mathbf{t}_e(e' \otimes_{\mathbb{E}} e'') = \top = \top \otimes_{\mathbb{E}} \top = \mathbf{t}_e(e') \otimes_{\mathbb{E}} \mathbf{t}_e(e'')$. □

Definition 29 ([14, Definition 7])**.** *Let $\mathbb{E}$ be a c-semiring. An element e of $\mathbb{E}$ is* collapsing *if there exist $e', e'' \in E$ such that $e' < e''$ and $e' \otimes e = e'' \otimes e$.*

Lemma 27 ([19, Lemma 1])**.** *Let $\mathbb{E}$ be a c-semiring and $e \in \mathbb{E}$. Then e is cancellative if and only if it is not collapsing.*

Proof. Assume that e is cancellative. Then for all $e_1, e_2 \in \mathbb{E}$ with $e_1 \neq e_2$, in particular those for which $e_1 < e_2$, it holds that $e_1 \otimes e \neq e_2 \otimes e$. It then follows that e cannot be collapsing. For the other direction, assume that e is not cancellative. Then there exist e_1, e_2 for which $e_1 \neq e_2$ and $e \otimes e_1 = e \otimes e_2$. If e_1 and e_2 are ordered by $<$ we are done immediately. Otherwise, choose $e_3 = e_1 \oplus e_2$. Immediately, we see that $e_1 \leq e_3$; it can also be seen that $e_1 \neq e_3$ (for otherwise e_1 and e_2 would be ordered by $<$); thus we have that $e_1 < e_3$. Moreover:

$$e_3 \otimes e = (e_1 \oplus e_2) \otimes e = (e_1 \otimes e) \oplus (e_2 \otimes e) = e_1 \otimes e$$

which establishes that e is collapsing. □

Lemma 28. *There exists a c-semiring that is totally ordered yet not cancellative.*

Proof. Consider the c-semiring $\mathbb{E} = \langle \{\emptyset, \{a\}, \{a, b\}\}, \bigcup, \cap, \emptyset, \{a, b\} \rangle$. This c-semiring is totally ordered, since $\emptyset \leq_{\mathbb{E}} \{a\} \leq_{\mathbb{E}} \{a, b\}$. Also, $\{a\} \cap \{a, b\} = \{a\} = \{a\} \cap \{a\}$, and therefore $\{a\}$ is not cancellative, making $\mathbb{E}$ a totally ordered yet non-cancellative c-semiring. □

Lemma 29 ([10, Theorem 57])**.** *Let $\mathbb{E}$ and $\mathbb{F}$ be c-semirings. The product c-semiring $\mathbb{E} \times \mathbb{F}$ is indeed a c-semiring.*

Proof. We verify the axioms of Definition 1 one-by-one.

- Let $\langle e, f \rangle \in \mathbb{E} \times \mathbb{F}$. Then we can compute

$$\begin{aligned} \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} \{\langle e, f \rangle\} &= \left\langle \bigvee\nolimits_{\mathbb{E}} \{e\}, \bigvee\nolimits_{\mathbb{F}} \{f\} \right\rangle = \langle e, f \rangle \\ \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} \emptyset &= \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(\emptyset), \bigvee\nolimits_{\mathbb{F}} \mathsf{Pr}_2(\emptyset) \right\rangle = \left\langle \bigvee\nolimits_{\mathbb{E}} \emptyset, \bigvee\nolimits_{\mathbb{F}} \emptyset \right\rangle = \langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}} \rangle = \mathbf{0}_{\mathbb{E}\times\mathbb{F}} \\ \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} \mathbb{E} \times \mathbb{F} &= \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(\mathbb{E} \times \mathbb{F}), \bigvee\nolimits_{\mathbb{F}} \mathsf{Pr}_2(\mathbb{E} \times \mathbb{F}) \right\rangle = \left\langle \bigvee\nolimits_{\mathbb{E}} \mathbb{E}, \bigvee\nolimits_{\mathbb{F}} \mathbb{F} \right\rangle = \langle \mathbf{1}_{\mathbb{E}}, \mathbf{1}_{\mathbb{F}} \rangle = \mathbf{1}_{\mathbb{E}\times\mathbb{F}} \end{aligned}$$

- Let $\mathcal{S} \subseteq 2^{\mathbb{E}\times\mathbb{F}}$. Then

$$\begin{aligned} \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} \bigcup \{S : S \in \mathcal{S}\} &= \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1\left(\bigcup\{S : S \in \mathcal{S}\}\right), \bigvee\nolimits_{\mathbb{F}} \mathsf{Pr}_2\left(\bigcup\{S : S \in \mathcal{S}\}\right) \right\rangle && \text{(def. } \textstyle\bigvee_{\mathbb{E}\times\mathbb{F}}) \\ &= \left\langle \bigvee\nolimits_{\mathbb{E}} \bigcup\{\mathsf{Pr}_1(S) : S \in \mathcal{S}\}, \bigvee\nolimits_{\mathbb{F}} \bigcup\{\mathsf{Pr}_2(S) : S \in \mathcal{S}\} \right\rangle && \text{(elementary)} \\ &= \left\langle \bigvee\nolimits_{\mathbb{E}} \left\{ \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S) : S \in \mathcal{S} \right\}, \bigvee\nolimits_{\mathbb{F}} \left\{ \bigvee\nolimits_{\mathbb{F}} \mathsf{Pr}_2(S) : S \in \mathcal{S} \right\} \right\rangle && \text{(flattening)} \\ &= \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} \left\{ \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S), \bigvee\nolimits_{\mathbb{F}} \mathsf{Pr}_2(S) \right\rangle : S \in \mathcal{S} \right\} && \text{(def. } \textstyle\bigvee_{\mathbb{E}\times\mathbb{F}}) \\ &= \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} \left\{ \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} S : S \in \mathcal{S} \right\} && \text{(def. } \textstyle\bigvee_{\mathbb{E}\times\mathbb{F}}) \end{aligned}$$

– Let $\langle e, f\rangle \in \mathbb{E} \times \mathbb{F}$. Then

$$\langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}}\rangle \otimes_{\mathbb{E}\times\mathbb{F}} \langle e, f\rangle = \langle \mathbf{0}_{\mathbb{E}} \otimes_{\mathbb{E}} e, \mathbf{0}_{\mathbb{F}} \otimes_{\mathbb{F}} f\rangle = \langle \mathbf{0}_{\mathbb{E}}, \mathbf{0}_{\mathbb{F}}\rangle$$

Similarly $\langle \mathbf{1}_{\mathbb{E}}, \mathbf{1}_{\mathbb{F}}\rangle \otimes_{\mathbb{E}\times\mathbb{F}} \langle e, f\rangle = \langle e, f\rangle$.

– Let $S \subseteq \mathbb{E} \times \mathbb{F}$ and $\langle e, f\rangle \in \mathbb{E}$. Then

$$\begin{aligned}
\langle e, f\rangle \otimes_{\mathbb{E}\times\mathbb{F}} \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} S &= \langle e, f\rangle \otimes_{\mathbb{E}\times\mathbb{F}} \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S), \bigvee\nolimits_{\mathbb{F}} \mathsf{Pr}_2(S)\right\rangle && \text{(def. } \textstyle\bigvee_{\mathbb{E}\times\mathbb{F}}) \\
&= \left\langle e \otimes_{\mathbb{E}} \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S), f \otimes_{\mathbb{F}} \bigvee\nolimits_{\mathbb{F}} \mathsf{Pr}_2(S)\right\rangle && \text{(def. } \otimes_{\mathbb{E}\times\mathbb{F}}) \\
&= \left\langle \bigvee\nolimits_{\mathbb{E}} \{e \otimes_{\mathbb{E}} e' : e' \in \mathsf{Pr}_1(S)\}, \bigvee\nolimits_{\mathbb{F}} \{f \otimes_{\mathbb{F}} f' : f' \in \mathsf{Pr}_2(S)\}\right\rangle && \text{(distributivity)} \\
&= \left\langle \bigvee\nolimits_{\mathbb{E}} \{\mathsf{Pr}_1(\langle e, f\rangle \otimes_{\mathbb{E}\times\mathbb{F}} s) : s \in S\}, \bigvee\nolimits_{\mathbb{F}} \{\mathsf{Pr}_2(\langle e, f\rangle \otimes_{\mathbb{E}\times\mathbb{F}s}) : s \in S\}\right\rangle && \text{(elementary)} \\
&= \bigvee\nolimits_{\mathbb{E}\times\mathbb{F}} \{\langle e, f\rangle \otimes_{\mathbb{E}\times\mathbb{F}} s : s \in S && \text{(def. } \textstyle\bigvee_{\mathbb{E}\times\mathbb{F}})
\end{aligned}$$

Lastly, it is easy to verify that $\otimes_{\mathbb{E}\times\mathbb{F}}$ is commutative and associative. □

Lemma 30 ([19, Lemma 2]). *Let $\mathbb{E}$ and $\mathbb{F}$ be c-semirings. If $\mathcal{S} \subseteq 2^{\mathbb{E}\times\mathbb{F}}$, then*

$$\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} \left(\bigcup\{S : S \in \mathcal{S}\}\right) = \bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} \left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}$$

Proof. By unrolling the definitions, we find that we essentially have to establish the following equalities:

$$\bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\bigcup\{S : S \in \mathcal{S}\}\right) = \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}\right) \tag{1}$$

$$\bigvee\nolimits_{\mathbb{F}} m \left(\bigcup\{S : S \in \mathcal{S}\}\right) = \bigvee\nolimits_{\mathbb{F}} m \left(\left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}\right) \tag{2}$$

To prove Equation 1, we use that

$$\mathsf{Pr}_1 \left(\left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}\right) = \left\{\bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S) : S \in \mathcal{S}\right\}$$

thus we can derive

$$\begin{aligned}
\bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\bigcup\{S : S \in \mathcal{S}\}\right) &= \bigvee\nolimits_{\mathbb{E}} \left(\bigcup\{\mathsf{Pr}_1(S) : S \in \mathcal{S}\}\right) && \text{(def. } \mathsf{Pr}_1) \\
&= \bigvee\nolimits_{\mathbb{E}} \left\{\bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S) : S \in \mathcal{S}\right\} && \text{(flattening } \textstyle\bigvee_{\mathbb{E}}) \\
&= \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}\right) && \text{(def. } \mathsf{Pr}_1 \text{ and } \textstyle\bigvee_{\mathbb{E}\rhd\mathbb{F}})
\end{aligned}$$

One can easily prove that for a c-semiring $\mathbb{E}$ with $E' \subseteq E \subseteq \mathbb{E}$, if $\bigvee_{\mathbb{E}} E \in E'$ then $\bigvee_{\mathbb{E}} E' = \bigvee_{\mathbb{E}} E$ $(*)$. We can then prove Equation 2 as follows:

$$\begin{aligned}
&\bigvee\nolimits_{\mathbb{F}} m \left(\left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}\right) \\
&= \bigvee\nolimits_{\mathbb{F}} \left\{e_2 : \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}\right), e_2\right\rangle \in \left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}\right\} && \text{(def. } m) \\
&= \bigvee\nolimits_{\mathbb{F}} \left\{\bigvee\nolimits_{\mathbb{F}} m(S') : S' \in \mathcal{S},\ \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\left\{\bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S : S \in \mathcal{S}\right\}\right) = \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S')\right\} && \text{(def. } \textstyle\bigvee_{\mathbb{E}\rhd\mathbb{F}}) \\
&= \bigvee\nolimits_{\mathbb{F}} \left\{\bigvee\nolimits_{\mathbb{F}} m(S') : S' \in \mathcal{S},\ \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\bigcup\{S : S \in \mathcal{S}\}\right) = \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S')\right\} && \text{(Equation 1)} \\
&= \bigvee\nolimits_{\mathbb{F}} \left\{e_2 : S' \in \mathcal{S},\ \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\bigcup\{S : S \in \mathcal{S}\}\right) = \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S'),\ e_2 \in m(S')\right\} && \text{(flattening } \textstyle\bigvee_{\mathbb{F}}) \\
&= \bigvee\nolimits_{\mathbb{F}} \left\{e_2 : S' \in \mathcal{S},\ \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\bigcup\{S : S \in \mathcal{S}\}\right), e_2\right\rangle \in S'\right\} && \text{(def. } m \text{ and } (*)) \\
&= \bigvee\nolimits_{\mathbb{F}} \left\{e_2 : \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1 \left(\bigcup\{S : S \in \mathcal{S}\}\right), e_2\right\rangle \in \bigcup\{S : S \in \mathcal{S}\}\right\} && \text{(def. } \textstyle\bigcup) \\
&= \bigvee\nolimits_{\mathbb{F}} m \left(\bigcup\{S : S \in \mathcal{S}\}\right) && \text{(def. } m)
\end{aligned}$$

Thus establishing the desired equalities. □

Lemma 31 ([19, Lemma 3]). *Let $\mathbb{E}$ and $\mathbb{F}$ be c-semirings such that $S \subseteq \mathbb{E} \rhd \mathbb{F}$ and $s \in \mathbb{E} \rhd \mathbb{F}$. Then*

$$s \otimes_{\mathbb{E}\rhd\mathbb{F}} \bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} S = \bigvee\nolimits_{\mathbb{E}\rhd\mathbb{F}} \{s \otimes_{\mathbb{E}\rhd\mathbb{F}} s' : s' \in S\}$$

Proof. Let $s = \langle e, f\rangle$ and $S' = \{s \otimes_{\mathbb{E}\triangleright\mathbb{F}} s' : s' \in S\}$. First, note that the following equality holds immediately by distributivity of $\otimes_{\mathbb{E}}$:

$$e_1 \otimes \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S) = \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S') \tag{1}$$

Unwinding the definitions and using Equation 1, we find the following for the left-hand side of the claim:

$$s \otimes_{\mathbb{E}\triangleright\mathbb{F}} \bigvee\nolimits_{\mathbb{E}\triangleright\mathbb{F}} S = \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S'), e_2 \otimes_{\mathbb{F}} \bigvee\nolimits_{\mathbb{F}} m(S) \right\rangle \tag{2}$$

Similarly, for the right-hand side we find that

$$\bigvee\nolimits_{\mathbb{E}\triangleright\mathbb{F}} \{s \otimes_{\mathbb{E}\triangleright\mathbb{F}} s' : s' \in S\} = \bigvee\nolimits_{\mathbb{E}\triangleright\mathbb{F}} S' = \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S'), \bigvee\nolimits_{\mathbb{F}} m(S') \right\rangle$$

Having established that the first components are equal, it remains to prove that this holds for the second component as well. If $e \in \overline{\mathcal{C}}(\mathbb{E})$ then $f = \mathbf{0}_{\mathbb{F}}$, thus $f \otimes_{\mathbb{F}} \bigvee_{\mathbb{F}} m(S) = \mathbf{0}_{\mathbb{F}}$. Moreover, for any element $\langle e', f'\rangle \in S'$ it must be that $f' = \mathbf{0}_{\mathbb{F}}$, thus $\bigvee_{\mathbb{F}} m(S') = \mathbf{0}_{\mathbb{F}}$.

The case where $e \in \mathcal{C}(\mathbb{E})$ remains. Let $m'(S) = \{f \otimes_{\mathbb{F}} f' : f' \in m(S)\}$. We can immediately rephrase the second component of Equation 2 as $\bigvee_{\mathbb{F}} m'(S)$. It now suffices to prove that $m'(S) = m(S')$. Let $f \otimes_{\mathbb{F}} f' \in m'(S)$. Then $\langle \bigvee_{\mathbb{E}} \mathsf{Pr}_1(S), f'\rangle \in S$, thus

$$s \otimes_{\mathbb{E}\triangleright\mathbb{F}} \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S), f' \right\rangle = \left\langle \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S'), f \otimes_{\mathbb{F}} f' \right\rangle \in S'$$

and therefore $f \otimes_{\mathbb{F}} f' \in m(S')$.

For the other direction, let $f \otimes_{\mathbb{F}} f' \in m(S')$. Then $\langle e \otimes_{\mathbb{E}} \bigvee_{\mathbb{E}} \mathsf{Pr}_1(S), f \otimes_{\mathbb{F}} f'\rangle \in S'$, thus there exists a $\langle e'', f''\rangle \in S$ such that

$$\langle e \otimes_{\mathbb{E}} f'', f \otimes_{\mathbb{F}} f''\rangle = \left\langle e \otimes_{\mathbb{E}} \bigvee\nolimits_{\mathbb{E}} \mathsf{Pr}_1(S), f \otimes_{\mathbb{F}} f' \right\rangle$$

Because $e \in \mathcal{C}(\mathbb{E})$, we can derive that $e'' = \bigvee_{\mathbb{E}} \mathsf{Pr}_1(S')$ and thus $f'' \in m(S')$, which implies that $f \otimes_{\mathbb{F}} f'' = f \otimes_{\mathbb{F}} f' \in m'(S')$. □

Theorem 5. *Let $\mathbb{E}$ and $\mathbb{F}$ be c-semirings. Then $\mathbb{E} \triangleright \mathbb{F}$ is indeed a c-semiring.*

Proof. The proofs that for $\langle e, f\rangle \in \mathbb{E} \triangleright \mathbb{F}$ we have that $\bigvee_{\mathbb{E}\triangleright\mathbb{F}}\{\langle e, f\rangle\} = \langle e, f\rangle$, as well as $\bigvee_{\mathbb{E}\triangleright\mathbb{F}} \emptyset = \mathbf{0}_{\mathbb{E}\triangleright\mathbb{F}}$ and $\bigvee_{\mathbb{E}\triangleright\mathbb{F}} \mathbb{E} \triangleright \mathbb{F} = \mathbf{1}_{\mathbb{E}\triangleright\mathbb{F}}$ are similar to Lemma 29, as are the proofs that $\otimes_{\mathbb{E}\triangleright\mathbb{F}}$ is commutative and associative, and that for $s \in \mathbb{E} \triangleright \mathbb{F}$ it holds that $s \otimes_{\mathbb{E}\triangleright\mathbb{F}} \mathbf{0}_{\mathbb{E}\triangleright\mathbb{F}} = \mathbf{0}_{\mathbb{E}\triangleright\mathbb{F}}$ and $s \otimes_{\mathbb{E}\triangleright\mathbb{F}} \mathbf{1}_{\mathbb{E}\triangleright\mathbb{F}} = s$. For the flattening principle, we refer to Lemma 30, and for distributivity of $\otimes_{\mathbb{E}\triangleright\mathbb{F}}$ over $\bigvee_{\mathbb{E}\triangleright\mathbb{F}}$, we refer to Lemma 31. □

Lemma 4 ([6, Theorem 2.1.4])**.** *Let $\mathbb{E}$ be a c-semiring. Then there exists a unique operator $\bigwedge : 2^{\mathbb{E}} \to \mathbb{E}$ such that $\langle \mathbb{E}, \leq, \bigvee, \bigwedge\rangle$ is a complete lattice.*

Proof. By Lemma 1, we already know that $\leq$ is a partial order. It remains to be shown that $\bigvee$ is indeed the least upper bound operators, and that we can find a suitable greatest lower bound operator $\bigwedge$. We begin with $\bigvee$. Let $E \subseteq \mathbb{E}$, and let $e \in \mathbb{E}$ such that for all $e' \in E$ it holds that $e' \leq e$ (i.e., e is an upper bound of E). Then for all $e' \in E$, $e' \leq \bigvee E \leq e$, because we can derive:

$$\begin{aligned}
e' \vee \bigvee E &= \bigvee(\{e'\} \cup E) = \bigvee E \\
e \vee \bigvee E &= \bigvee(\{e\} \cup E) \\
&= \bigvee \left(\bigcup\nolimits_{e'' \in E} \{e, e''\}\right) \\
&= \bigvee\{e \vee e'' : e'' \in E'\} = \bigvee\{e\} = e
\end{aligned}$$

We now choose $\bigwedge$ to be the operator that takes the least upper bound of all lower bounds of the input, i.e.,

$$\bigwedge E = \bigvee\{b \in E : \forall e' \in E. b \leq e'\}$$

To show that $\bigwedge$ is the greatest lower bound operator, let $e \in \mathbb{E}$ and $E \subseteq \mathbb{E}$ such that for all $e' \in E$ we have that $e \leq e'$. Then $e \leq \bigwedge E$, because we can derive as follows

$$\begin{aligned}
e \vee \bigwedge E &= \bigvee\left(\{e\} \cup \{b \in \mathbb{E},\ \forall e' \in E. b \leq e'\}\right) \\
&= \bigvee\{b \in E : \forall e' \in E. b \leq e'\} = \bigwedge E
\end{aligned}$$

Moreover, for all $e' \in E$ it holds that $\bigwedge E \leq e'$, since

$$\begin{aligned} e' \vee \bigwedge E &= \bigvee \{e' \vee b : \forall e'' \in E. b \leq e''\} \\ &= \bigvee \{e'\} = e' \end{aligned}$$

Thus for all $e' \in E$, $e \leq \bigwedge E \leq e'$. We can therefore conclude that $\langle \mathbb{E}, \leq, \bigvee, \bigwedge \rangle$ is a complete lattice.

To see that $\bigwedge$ is unique, let $\circledast : 2^{\mathbb{E}} \to \mathbb{E}$ also be a greatest lower bound operator. Then for all $E \subseteq \mathbb{E}$ we can derive that $\circledast E \leq \bigwedge E \leq \circledast E$, thus $\circledast E = \bigwedge E$ by antisymmetry of $\leq$, and therefore $\circledast = \bigwedge$. □

Lemma 32. *Let $\mathbb{E}$ and $\mathbb{F}$ be cancellative c-semirings. Then $\mathbb{E} \odot \mathbb{F}$ is indeed a c-semiring.*

Proof. We prove that $\mathbb{E} \odot \mathbb{F}$ is closed under the operators of Definition 7; all other properties can be verified similar to Lemma 29.

Let $S \subseteq \mathbb{E} \odot \mathbb{F}$. Then we know that $\mathsf{Pr}_1(S) \subseteq \mathcal{C}(\mathbb{E}) \cup \{\mathbf{0}_{\mathbb{E}}\} = \mathbb{E}$, thus $\bigvee_{\mathbb{E}} \mathsf{Pr}_1(S) \in \mathbb{E}$. If, however, $\bigvee_{\mathbb{E}} \mathsf{Pr}_1(S) = \mathbf{0}_{\mathbb{E}}$, then $\mathsf{Pr}_1(S)$ is empty or $\mathsf{Pr}_1(S) = \{\mathbf{0}_{\mathbb{E}}\}$. In the former case, we know that $\mathsf{Pr}_2(S)$ is empty and in the latter case, we know that $\mathsf{Pr}_2(S) = \{\mathbf{0}_{\mathbb{F}}\}$ (otherwise S would not be a subset of $\mathbb{E} \odot \mathbb{F}$) — thus, in both cases we know that $\bigvee_{\mathbb{F}} \mathsf{Pr}_2(S) = \mathbf{0}_{\mathbb{F}}$. We can therefore conclude that, in the case where $\bigvee_{\mathbb{E}} \mathsf{Pr}_1(S) = \mathbf{0}_{\mathbb{E}}$ (or, by symmetry, $\bigvee_{\mathbb{F}} \mathsf{Pr}_2(S) = \mathbf{0}_{\mathbb{F}}$), it holds that $\bigvee_{\mathbb{E} \odot \mathbb{F}} S \in \mathbb{E} \odot \mathbb{F}$. The case remains where $\bigvee_{\mathbb{E}} \mathsf{Pr}_1(S) \in \mathcal{C}(\mathbb{E})$ and $\bigvee_{\mathbb{F}} \mathsf{Pr}_2(S) \in \mathcal{C}(\mathbb{F})$. But then $\bigvee_{\mathbb{E} \odot \mathbb{F}} S \in \mathbb{E} \odot \mathbb{F}$ by definition of $\bigvee_{\mathbb{E} \odot \mathbb{F}}$.

For closure of $\mathbb{E} \odot \mathbb{F}$ under $\otimes_{\mathbb{E} \odot \mathbb{F}}$, we observe that if $e_1, e_2 \in \mathcal{C}(\mathbb{E})$, then $e \otimes_{\mathbb{E}} e' \in \mathcal{C}(\mathbb{E})$, too. To see this, take $e_3, e_4 \in \mathcal{C}(\mathbb{E})$ such that $(e_1 \otimes_{\mathbb{E}} e_2) \otimes_{\mathbb{E}} e_3 = (e_1 \otimes_{\mathbb{E}} e_2) \otimes_{\mathbb{E}} e_4$; by associativity of $\otimes_{\mathbb{E}}$, we know that $e_1 \otimes_{\mathbb{E}} (e_2 \otimes_{\mathbb{E}} e_3) = e_1 \otimes_{\mathbb{E}} (e_2 \otimes_{\mathbb{E}} e_4)$; by cancellativity of e_1 it follows that $e_2 \otimes_{\mathbb{E}} e_3 = e_2 \otimes_{\mathbb{E}} e_4$, and by cancellativity of e_2 we have that $e_3 = e_4$. By symmetry, a similar argument holds for $\mathbb{F}$. Thus, if $\langle e, f \rangle, \langle e', f' \rangle \in \mathbb{E} \odot \mathbb{F}$, then if $\langle e, f \rangle = \mathbf{0}_{\mathbb{E} \odot \mathbb{F}}$ or if $\langle e', f' \rangle = \mathbf{0}_{\mathbb{E} \odot \mathbb{F}}$, we have that $\langle e, f \rangle \otimes_{\mathbb{E} \odot \mathbb{F}} \langle e', f' \rangle = \mathbf{0}_{\mathbb{E} \odot \mathbb{F}} \in \mathbb{E} \odot \mathbb{F}$. If this is not the case, then $e, e' \in \mathcal{C}(\mathbb{E})$ and $f, f' \in \mathcal{C}(\mathbb{F})$, thus $\langle e, f \rangle \otimes_{\mathbb{E} \odot \mathbb{F}} \langle e, f \rangle = \langle e \otimes_{\mathbb{E}} e', f \otimes_{\mathbb{F}} f' \rangle \in \mathcal{C}(\mathbb{E}) \times \mathcal{C}(\mathbb{F}) \subseteq \mathbb{E} \odot \mathbb{F}$. □

Lemma 33. *Let $\mathbb{E}$ and $\mathbb{F}$ be cancellative c-semirings. Then $\mathbb{E} \odot \mathbb{F}$ is cancellative.*

Proof. Let $\langle e_1, f_1 \rangle, \langle e_2, f_2 \rangle, \langle e_3, f_3 \rangle \in \mathbb{E} \odot \mathbb{F}$ be such that $\langle e_1, f_1 \rangle \otimes_{\mathbb{E} \odot \mathbb{F}} \langle e_2, f_2 \rangle = \langle e_1, f_1 \rangle \otimes_{\mathbb{E} \odot \mathbb{F}} \langle e_3, f_3 \rangle$, with $\langle e_1, f_1 \rangle \neq \mathbf{0}_{\mathbb{E} \odot \mathbb{F}}$. Then we know that $e_1 \in \mathcal{C}(\mathbb{E})$ and $f_1 \in \mathcal{C}(\mathbb{F})$. Also, by definition of $\otimes_{\mathbb{E} \odot \mathbb{F}}$, we have that $e_1 \otimes_{\mathbb{E}} e_2 = e_1 \otimes_{\mathbb{E}} e_3$ and $f_1 \otimes_{\mathbb{F}} f_2 = f_1 \otimes_{\mathbb{F}} f_3$. Since $e_1 \in \mathcal{C}(\mathbb{E})$ and $f_1 \in \mathcal{C}(\mathbb{F})$, we have that $e_2 = e_3$ and $f_2 = f_3$, thus $\langle e_2, f_2 \rangle = \langle e_3, f_3 \rangle$. As a consequence, $\langle e_1, f_1 \rangle$ is cancellative for all $\langle e_1, f_1 \rangle \in \mathbb{E} \odot \mathbb{F}$ with $\langle e_1, f_1 \rangle \neq \mathbf{0}_{\mathbb{E} \odot \mathbb{F}}$, thus $\mathbb{E} \odot \mathbb{F}$ is cancellative. □

B Proofs for Subsection 7.3

B.1 Proofs for the generalized pointwise order

Lemma 34. *Let $\mathbb{E}$ be a c-semiring. If $w, x, y, z \in \mathbb{E}^*$ with $|w| = n = |y|$ and $|x| = m = |z|$, then $w \cdot x \vee_{\mathbb{E}^{n+m}} y \cdot z = (w \vee_{\mathbb{E}^n} y) \cdot (x \vee_{\mathbb{E}^m} z)$ and $w \cdot x \wedge_{\mathbb{E}^n} y \cdot z = (w \wedge_{\mathbb{E}^m} y) \cdot (x \wedge_{\mathbb{E}^m} z)$.*

Proof. For the first equality, observe that the p-th position of $w \cdot x \vee_{\mathbb{E}^{n+m}} y \cdot z$ is equal to the p-th position of $w \vee x$ when $p \leq n$ and to the $(p-n)$-th position of $y \vee z$ otherwise. At any rate, it is equal to the p-th position of $(w \vee_{\mathbb{E}^n} x) \cdot (y \vee_{\mathbb{E}^m} z)$. The proof of the second equality is similar. □

Lemma 35. *Let $\mathbb{E}$ be a c-semiring. If $w, x, y, z \in \mathbb{E}^*$ with $|w| = n = |y|$ and $|x| = m = |z|$, then when $w \leq_{\mathbb{E}^n} y$ and $x \leq_{\mathbb{E}^m} z$ it follows that $w \cdot x \leq_{\mathbb{E}^{n+m}} y \cdot z$. Moreover, when $y \leq_{\mathbb{E}^n} w$ and $z \leq_{\mathbb{E}^m} y$, it follows that $w \cdot x \leq_{\mathbb{E}^{n+m}} y \cdot z$.*

Proof. By application of Lemma 34, we have that $w \cdot x \vee_{\mathbb{E}^{n+m}} y \cdot z = (w \vee_{\mathbb{E}^n} y) \cdot (x \vee_{\mathbb{E}^m} z) = y \cdot z$, thus $w \cdot x \leq_{\mathbb{E}^{n+m}} y \cdot z$. The proof of the second claim is similar. □

Lemma 19. *Let $A = \langle Q, \Sigma, \mathbb{E}, \to, q^0, t \rangle$ be an SCA. If $w, x \in \mathfrak{P}(A)^*$ such that $|w| = n = |x|$ and $w \leq_{\mathbb{E}^n} x$, then we have that*

$$\begin{gathered} \sharp_A(w) \leq_{\mathbb{E}^{n+1}} \sharp_A(x) \\ \flat_A(x) \leq_{\mathbb{E}^{n+1}} \flat_A(w) \end{gathered}$$

Proof. We prove the first claim by induction on $|w|$. If $|w| = \epsilon$ the claim holds immediately. Now let $w = y \cdot e$ and $x = z \cdot f$ with $w \leq x$. Then $y \leq_{\mathbb{E}^{n-1}} z$ and $e \leq f$. Assuming that the claim holds for y and z we derive

$$\begin{aligned} \sharp_A(y \cdot e) &= \sharp_A(y) \cdot e \vee_{\mathbb{E}^{n+1}} w \cdot e \cdot t && \text{(Definition 26)} \\ &\leq_{\mathbb{E}^{n+1}} \sharp_A(z) \cdot f \vee_{\mathbb{E}^{n+1}} z \cdot f \cdot t && \text{(Induction hypothesis, Lemma 35)} \end{aligned}$$

$$= \sharp_A(z \cdot f) \tag{Definition 26}$$

The proof of the second claim is similar. □

Lemma 36. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA and let $n \in \mathbb{N}$. If $n \in \mathbb{N}$, then $\sharp_A(t^n) = t^{n+1} = \flat_A(t^n)$.*

Proof. We prove the first equality by induction on n. If $n = 0$, then the claim holds immediately by Definition 26. Assume now that the claim holds for n and derive

$$\sharp_A(t^{n+1}) = \sharp_A(t^n) \cdot t \vee_{\mathbb{E}^n} t^{n+1} \cdot t = t^{n+1} \cdot t \vee_{\mathbb{E}^n} t^{n+1} \cdot t = t^{n+2}$$

The proof of the second equality is similar. □

Lemma 37. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA and let $m, n \in \mathbb{N}$. Then $\sharp_A^m(t^n) = t^{m+n} = \flat_A^m(t^n)$.*

Proof. We prove the first equality by induction on m. If $m = 0$, the equality holds immediately. Assume now the equality holds for m and derive

$$\sharp_A^{m+1}(t^n) = \sharp_A(\sharp_A^m(t^n)) = \sharp_A(t^{m+n}) = t^{m+n+1}$$

In which the second step follows from Lemma 36. The proof of the second claim is similar. □

Lemma 38. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA. If $w, x \in \mathfrak{P}(A)^*$ with $|w| = n = |x|$, then*

$$\begin{aligned} \sharp_A(w \vee_{\mathbb{E}^n} x) &= \sharp_A(w) \vee_{\mathbb{E}^{n+1}} \sharp_A(x) \\ \flat_A(w \wedge_{\mathbb{E}^n} x) &= \flat_A(w) \wedge_{\mathbb{E}^{n+1}} \flat_A(x) \end{aligned}$$

Proof. We prove the first claim by induction on $|w|$. If $w = \epsilon$, then

$$\sharp_A(\epsilon \vee_{\mathbb{E}^n} \epsilon) = \sharp_A(\epsilon) = t = t \vee_{\mathbb{E}^{n+1}} t = \sharp_A(\epsilon) \vee_{\mathbb{E}^{n+1}} \sharp_A(\epsilon)$$

Now let $w = y \cdot e$ and $x = z \cdot f$ and assume that the claim holds for x and z. Then

$$\begin{aligned} \sharp_A(y \cdot e \vee_{\mathbb{E}^n} z \cdot f) &= \sharp_A((y \vee_{\mathbb{E}^{n-1}} z) \cdot (e \vee f)) && \text{(Lemma 34)} \\ &= \sharp_A(y \vee_{\mathbb{E}^{n-1}} z) \cdot (e \vee f) \vee (y \vee_{\mathbb{E}^{n-1}} z) \cdot (e \vee f) \cdot t && \text{(Definition 26)} \\ &= (\sharp_A(y) \vee_{\mathbb{E}^{n-1}} \sharp_A(z)) \cdot (e \vee f) \vee_{\mathbb{E}^{n+1}} (y \vee_{\mathbb{E}^{n-1}} z) \cdot (e \vee f) \cdot t && \text{(Induction hypothesis)} \\ &= \sharp_A(y) \cdot e \vee_{\mathbb{E}^{n+1}} y \cdot e \cdot t \vee_{\mathbb{E}^{n+1}} \sharp_A(z) \cdot f \vee_{\mathbb{E}^{n+1}} y \cdot f \cdot t && \text{(Lemma 34)} \\ &= \sharp_A(y \cdot e) \vee_{\mathbb{E}^{n+1}} \sharp_A(z \cdot f) = \sharp_A(w) \vee_{\mathbb{E}^{n+1}} \sharp_A(x) && \text{(Definition 26)} \end{aligned}$$

The proof of the second claim is similar. □

Lemma 39. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be an SCA. If $w \cdot e \in \mathfrak{P}(A)^*$, then for all $n \in \mathbb{N}$ it holds that*

$$\begin{aligned} \sharp_A^n(w) \cdot e &\leq_{\mathbb{E}^{|w|+n}} \sharp_A^n(w \cdot e) \\ \flat_A^n(w \cdot e) &\leq_{\mathbb{E}^{|w|+n}} \flat_A^n(w) \cdot e \end{aligned}$$

Proof. We prove the first claim by induction on n. If $n = 0$ then the claim holds vacuously. For $n = 1$ we derive

$$\sharp_A(w) \cdot e \leq_{\mathbb{E}^{|w|+n}} \sharp_A(w) \cdot e \vee_{\mathbb{E}^{|w|+n}} w \cdot e \cdot t = \sharp_A(w \cdot e)$$

Now assume that the claim holds for all $n' \leq n$ and derive:

$$\begin{aligned} \sharp_A^{n+1}(w) \cdot e &= \sharp_A^n(\sharp_A(w)) \cdot e \\ &\leq_{\mathbb{E}^{|w|+n+1}} \sharp_A^n(\sharp_A(w) \cdot e) && \text{(Induction hypothesis)} \\ &\leq_{\mathbb{E}^{|w|+n+1}} \sharp_A^{n+1}(w \cdot e) && \text{(Induction hypothesis, Lemma 19)} \end{aligned}$$

The proof of the second claim is similar. □

Lemma 40. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$. If $w \in \mathfrak{P}(A)^*$, then for all $n \in \mathbb{N}$ it holds that*

$$\begin{aligned} \sharp_A^n(w) \cdot t &\leq_{\mathbb{E}^{n+|w|+1}} \sharp_A^{n+1}(w) \\ \flat_A^{n+1}(w) &\leq_{\mathbb{E}^{n+|w|+1}} \flat_A^n(w) \cdot t \end{aligned}$$

Proof. We prove the first claim by induction on n. For the basis, where $n = 0$, we distinguish two cases. If $w = \epsilon$, we find that $\sharp_A^0(\epsilon) \cdot t = \epsilon \cdot t = t = \sharp_A(\epsilon)$. If $w = x \cdot e$, we simply derive

$$\sharp_A^0(x \cdot e) \cdot t = x \cdot e \cdot t \leq_{\mathbb{E}^{n+|w|+1}} \sharp_A(x) \cdot e \vee_{\mathbb{E}^{n+|w|+1}} x \cdot e \cdot t = \sharp_A^1(x \cdot e)$$

Now assume that the claim holds for n and all w and observe that

$$\sharp_A^{n+1}(w) \cdot t = \sharp_A^n(\sharp_A(w)) \cdot t \leq_{\mathbb{E}^{n+|w|+1}} \sharp_A^{n+1}(\sharp_A(w)) = \sharp_A^{n+2}(w)$$

In which we use the induction hypothesis and Lemma 19 in the second step. The proof of the second claim is similar. □

Lemma 41. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be an SCA. If $n \in \mathbb{N}$ such that $n \geq 1$, $w \in \mathfrak{P}(A)^*$ and $e \in \mathfrak{P}(A)$, then*

$$\sharp_A^n(w \cdot e) = \sharp_A^n(w) \cdot e \vee_{\mathbb{E}^{n+|w|}} \sharp_A^{n-1}(w \cdot e) \cdot t$$
$$\flat_A^n(w \cdot e) = \flat_A^n(w) \cdot e \wedge_{\mathbb{E}^{n+|w|}} \flat_A^{n-1}(w \cdot e) \cdot t$$

Proof. Let $w \cdot e \in \mathfrak{P}(A)^*$. We prove first the claim by induction on n. For the base case, where $n = 1$, the claim is true by definition of $\sharp_A$. Now assume the claim is true for n and derive

$$\begin{aligned}
\sharp_A^{n+1}(w \cdot e) &= \sharp_A(\sharp_A^n(w \cdot e)) \\
&= \sharp_A(\sharp_A^n(w) \cdot e \vee_{\mathbb{E}^{n+|w|}} \sharp_A^{n-1}(w \cdot e) \cdot t) && \text{(Induction hypothesis)} \\
&= \sharp_A(\sharp_A^n(w) \cdot e) \vee_{\mathbb{E}^{n+|w|}} \sharp_A(\sharp_A^{n-1}(w \cdot e) \cdot t) && \text{(Lemma 38)} \\
&= \sharp_A(\sharp_A^n(w)) \cdot e \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A^n(w) \cdot e \cdot t \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A(\sharp_A^{n-1}(w \cdot e)) \cdot t \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A^{n-1}(w \cdot e) \cdot t \cdot t \\
& && \text{(Definition 26)} \\
&= \sharp_A^{n+1}(w) \cdot e \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A^n(w) \cdot e \cdot t \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A^n(w \cdot e) \cdot t \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A^{n-1}(w \cdot e) \cdot t \cdot t \\
&= \sharp_A^{n+1}(w) \cdot e \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A^n(w \cdot e) \cdot t \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A^{n-1}(w \cdot e) \cdot t \cdot t && \text{(Lemma 39)} \\
&= \sharp_A^{n+1}(w) \cdot e \vee_{\mathbb{E}^{n+|w|+1}} \sharp_A^n(w \cdot e) \cdot t && \text{(Lemma 40)}
\end{aligned}$$

The proof of the second claim is similar. □

Lemma 42. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA. If $w \in \mathfrak{P}(A)^*$ and $n > |w|$, then*

$$\mathsf{idle}_A(w, n) = \{y \cdot e : w = x \cdot e, y \in \mathsf{idle}_A(x, n-1)\} \cup \{x \cdot t : x \in \mathsf{idle}_A(w, n-1)\}$$

Proof. First, observe that since $w \in \mathfrak{P}(A)^*$ and A is threshold-free, we know that $w = \mathfrak{p}_A(w)$. To show that $w' \in \mathsf{idle}_A(w, n)$ it therefore suffices to show that $\mathfrak{p}_A(w') = w$ and $|w'| = n$.

We begin by proving the inclusion left-to-right. Let $w' \in \mathsf{idle}_A(w, n)$. If $w' = x \cdot t$, then $\mathfrak{p}_A(x) = \mathfrak{p}_A(w') = w$, and since $|x| = n-1$, we have that $x \in \mathsf{idle}_A(w, n-1)$. Therefore $w' \in \{x \cdot t : x \in \mathsf{idle}_A(w, n-1)\}$. The case remains where $w' = y \cdot e$ for $e \in \mathfrak{P}(A)$. Let $w = x \cdot e'$ for $e' \in \mathfrak{p}_A(E)$, then $y \cdot e = w' = \mathfrak{p}_A(w) = \mathfrak{p}_A(x \cdot e') = \mathfrak{p}_A(x) \cdot \mathfrak{p}_A(e')$, and since $e' \neq t$, it follows that $\mathfrak{p}_A(x) = y$; together with $|y| = n - 1$ we conclude that $y \in \mathsf{idle}_A(x, n-1)$ and thus $w' \in \{y \cdot e : w = x \cdot e, y \in \mathsf{idle}_A(x, n-1)\}$.

For the right-to-left inclusion, let $w' \in \{y \cdot e : w = x \cdot e, y \in \mathsf{idle}_A(x, n-1)\} \cup \{x \cdot t : x \in \mathsf{idle}_A(w, n-1)\}$. If $w' = x \cdot t$ for $x \in \mathsf{idle}_A(w, n-1)$, then $\mathfrak{p}_A(w') = \mathfrak{p}_A(x \cdot t) = \mathfrak{p}_A(x) = w$. With $|w'| = n$ it follows that $w' \in \mathsf{idle}_A(w, n)$. If on the other hand $w' = y \cdot e$ for $w = x \cdot e$ and $y \in \mathsf{idle}_A(x, n-1)$, then $\mathfrak{p}_A(w') = \mathfrak{p}_A(y) \cdot \mathfrak{p}_A(e) = x \cdot e = w$; together with $|w'| = n$ it follows that $w' \in \mathsf{idle}_A(w, n)$. □

Lemma 20. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA. Then for $w \in \mathbb{E}^*$ and $n \in \mathbb{N}$ with $|w| \leq n$, the following equalities hold:*

$$\bigvee \mathsf{idle}_A(w, n) = \sharp_A^{n-|w|}(w)$$
$$\bigwedge \mathsf{idle}_A(w, n) = \flat_A^{n-|w|}(w)$$

Proof. We prove the first equality by induction on $(|w|, n)$, ordered lexicographically. For this, our base cases include those where $n = |w|$ and where $|w| = 0$. For the case where $|w| = 0$, observe that $\mathsf{idle}_A(\epsilon, n) = \{t^n\}$, and conclude that $\mathsf{idle}_A(\epsilon, n) = \sharp_A^n(\epsilon)$ by Lemma 37. For the cases where $n = |w|$, first observe that $\mathsf{idle}_A(w, |w|) = \{w\}$. Now we can derive that

$$\bigvee \mathsf{idle}_A(w, |w|) = w = \sharp_A^0(w)$$

Assume now that the claim holds for n' and w' with $(|w'|, n')$ lexicographically smaller than $(|w|, n)$, and derive:

$$\bigvee\nolimits_{\mathbb{E}^n} \mathsf{idle}_A(z \cdot e, n) = \bigvee\nolimits_{\mathbb{E}^n} (\{y \cdot e : y \in \mathsf{idle}_A(z, n-1)\} \cup \{x \cdot t : x \in \mathsf{idle}_A(z \cdot e, n-1)\}) \qquad \text{(Lemma 42)}$$

$$\begin{aligned}
&= \bigvee\nolimits_{\mathbb{E}^n} \{y \cdot e : y \in \mathsf{idle}_A(z, n-1)\} \vee_{\mathbb{E}^n} \bigvee\nolimits_{\mathbb{E}^n} \{x \cdot t : x \in \mathsf{idle}_A(z \cdot e, n-1)\} && \text{(Flattening)} \\
&= \Big(\bigvee\nolimits_{\mathbb{E}^{n-1}} \mathsf{idle}_A(z, n-1) \cdot e\Big) \vee_{\mathbb{E}^n} \Big(\bigvee\nolimits_{\mathbb{E}^{n-1}} \mathsf{idle}_A(z \cdot e, n-1) \cdot t\Big) && \text{(Lemma 34)} \\
&= \sharp_A^{n-|w|}(z) \cdot e \vee_{\mathbb{E}^n} \sharp_A^{n-|w|-1}(z \cdot e) \cdot t && \text{(Induction hypothesis)} \\
&= \sharp_A^{n-|w|}(z \cdot e) && \text{(Lemma 41)} \\
&= \sharp_A^{n-|w|}(w)
\end{aligned}$$

The proof of the second claim is similar. □

B.2 Proofs for the generalized lexicographic order

Lemma 21. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA and let $n \in \mathbb{N}$. The relation $\sqsubseteq_A^n$ is reflexive.*

Proof. This follows immediately from the definition. Let $w \in \mathfrak{P}(A)^*$ such that $|w| \leq_{\mathbb{E}^*} n$. Since for any $w' \in \mathsf{idle}_A(w, n)$, we immediately have that $w' \leq_{\mathbb{E}^*} w'$, and thus $w' \sqsubseteq_A^n w'$. □

Lemma 22. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA, and let $n \in \mathbb{N}$. If $w, x \in \mathfrak{P}(A)^*$ are such that $n = \max(|w|, |x|)$ and $w \sqsubseteq_A^n x \sqsubseteq_A^n w$, then $w = x$.*

Proof. If $n = |x|$, then $\mathsf{idle}_A(x, n) = \{x\}$. Thus we know that for every $w' \in \mathsf{idle}_A(w, n)$ it holds that $w' \leq_{\mathbb{E}^*} x$. Moreover, we know that there exists a $w'' \in \mathsf{idle}_A(w, n)$ such that $x \leq_{\mathbb{E}^*} w''$. Since the former assertion also holds for w'', we know that $w'' \leq_{\mathbb{E}^*} x \leq_{\mathbb{E}^*} w''$ and therefore $w'' = x$ by antisymmetry of $\leq_{\mathbb{E}^*}$. But then w'' does not contain any occurrence of t (since x does not), so $|w| = n$ and thus $w'' = w$, allowing us to conclude that $w = x$.

For the case where $n = |w|$ a similar argument holds. Here, we know that $\mathsf{idle}_A(w, n) = \{w\}$, and there exists a $x' \in \mathsf{idle}_A(x, n)$ such that $w \leq_{\mathbb{E}^*} x'$. Moreover, for all $x'' \in \mathsf{idle}_A(x, n)$ it holds that $x'' \leq_{\mathbb{E}^*} w$. Since the latter assertion holds for x' as well, we have that $w \leq_{\mathbb{E}^*} x' \leq_{\mathbb{E}^*} w$. Therefore, $w = x'$ and it follows that x' does not contain any occurrence of i, thus $x' = x$, leading to a conclusion that $w = x$. □

Lemma 23. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA, and let $n \in \mathbb{N}$. The relation $\sqsubseteq_A^n$ is transitive.*

Proof. Let $w, x, y \in \mathfrak{P}(A)^*$ such that $|w|, |x|, |y| \leq_{\mathbb{E}^*} n$ and $w \sqsubseteq_A^n x \sqsubseteq_A^n y$. We need to show that $w \sqsubseteq_A^n y$. Let $w' \in \mathsf{idle}_A(w, n)$. Then by $w \sqsubseteq_A^n x$ there exists an $x' \in \mathsf{idle}_A(x, n)$ such that $w' \leq_{\mathbb{E}^*} x'$. For this particular x', by $x \sqsubseteq_A^n y$ we can find a $y' \in \mathsf{idle}_A(y, n)$ such that $x' \leq_{\mathbb{E}^*} y'$. Therefore, $w' \leq_{\mathbb{E}^*} y'$ by transitivity of $\leq_{\mathbb{E}^*}$. Since such a y' can be found for any $x' \in \mathsf{idle}_A(w, n)$, we have established that $w \sqsubseteq_A^n y$. □

Lemma 24. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t\rangle$ be a threshold-free SCA. If $w, x \in \mathfrak{P}(A)^*$ and $n \in \mathbb{N}$ such that $|w|, |x| \leq n$, then $w \sqsubseteq_A^n x$ if and only if $w \sqsubseteq_A^{n+1} x$.*

Proof. For the claim from left to right, let $w_2 \in \mathsf{idle}_A(w, n+1)$. We need to find an $x_2 \in \mathsf{idle}_A(x, n+1)$ such that $w_2 \leq_{\mathbb{E}^*} x_2$. Choose a k such that the k-th position of w_2 is t. Such a k always exists, since $n + 1 > |w|$. Remove this position from k to obtain $w_1 \in \mathsf{idle}_A(w, n)$. By the premise, there exists an $x_1 \in \mathsf{idle}_A(x, n)$ such that $w_1 \leq_{\mathbb{E}^*} x_1$. Insert a t at the k-th position of x_1 to obtain $x_2 \in \mathsf{idle}_A(x, n+1)$. Then $w_2 \leq_{\mathbb{E}^*} x_2$, because we inserted the same character into the same position of w_1 and x_1 to obtain w_2 respectively x_2. We conclude that $w \sqsubseteq_A^{n+1} x$.

For the claim from right to left, let $w_1 \in \mathsf{idle}_A(w, n)$. We need to find an $x_1 \in \mathsf{idle}_A(x, n)$ such that $w_1 \leq_{\mathbb{E}^*} x_1$. Choose $w_2 = t \cdot w_1$. Then $w_2 \in \mathsf{idle}_A(w, n+1)$, thus by the premise there exists a $x_2 \in \mathsf{idle}_A(x, n+1)$ such that $w_2 \leq_{\mathbb{E}^*} x_2$. The remainder of the proof is a case analysis.

For the first and easiest case, assume that x_2 starts with t. Write $x_2 = t \cdot x_1$. Since $t \cdot w_1 \leq_{\mathbb{E}^*} t \cdot x_1$, it follows that $w_1 \leq_{\mathbb{E}^*} x_1$. Moreover, $x_1 \in \mathsf{idle}_A(x, n)$, since $\mathfrak{p}_A(x_1) = h(t \cdot x_1) = \mathfrak{p}_A(x_2) = x$.

If, on the other hand, x_2 does not start with t, then we immediately know that $x \neq \epsilon$. Write e for the first character of x and choose $x_1 = x \cdot t^{n-|x|}$. First, observe that $x_1 \in \mathsf{idle}_A(x, n)$. We also know that $t < e$, since $t \cdot w_1 \leq_{\mathbb{E}^*} x_2$ and x_2 does not start with t (and must therefore start with $e \neq t$). It remains to show that $w_1 \leq_{\mathbb{E}^*} x_1$. We can ignore the case where w_1 starts with t, for then have $w_1 \leq_{\mathbb{E}^*} x_1$ and are done immediately.

Let us restate our premises in the final remaining case. We know that neither w_1 nor x_1 starts with t (and therefore both w and x are non-empty), and that $t \leq_{\mathbb{E}} e$, where e is the first character of x (and therefore of x_1). Consider $w_3 = w_1 \cdot t$. Also by our premise, there exists an $x_3 \in \mathsf{idle}_A(x, n+1)$ such that $w_3 \leq_{\mathbb{E}^*} x_3$. If x_3 does not start with t, then the first character of w_1 precedes e and we have $w_1 \leq_{\mathbb{E}^*} x_1$. If x_3 does start with t, then the first character of w_1 precedes t in $\leq_{\mathbb{E}}$, which in turn precedes e and we have $w_1 \leq_{\mathbb{E}^*} x_1$ again. □

C Proofs for Subsection 7.4

In this appendix, we work towards a proof of Theorem 3 and Theorem 4. The proof for these theorems is somewhat involved, and requires that we develop some auxiliary lemma's on $\leq_{\mathbb{E}^*}$ and $\sqsubseteq_A$ for c-semirings $\mathbb{E}$ and threshold-free automata A. To abbreviate notation, we fix the symbol $\mathbb{E}$ for any c-semiring, and A for any threshold-free SCA with underlying c-semiring $\mathbb{E}$.

C.1 Lemma's for the lexicographic order

We begin by proving some helpful lemma's on the lexicographic order

Lemma 43. *Let $w, x \in \mathbb{E}^*$ such that w is a prefix of x. Then $w \leq_{\mathbb{E}^*} x$.*

Proof. Write $x = w \cdot y$. Then $\epsilon \leq_{\mathbb{E}^*} y$, and consequently $w \leq_{\mathbb{E}^*} w \cdot y = x$. □

Lemma 44. *Let $w, x \in \mathbb{E}^*$. If $w <_{\mathbb{E}^*} x$ and w is not a prefix of x, then x is not a prefix of w*

Proof. Suppose towards a contradiction that x is a prefix of w. Then $x \leq_{\mathbb{E}^*} w$ by Lemma 43. If $x = w$ then $w <_{\mathbb{E}^*} w$; if $x <_{\mathbb{E}^*} w$ then $x <_{\mathbb{E}^*} w <_{\mathbb{E}^*} x$; both are contradictions. □

Lemma 45. *Let $w, x, y \in \mathbb{E}^*$. If $w <_{\mathbb{E}^*} x$ and w is not a prefix of x, then $w \cdot y <_{\mathbb{E}^*} x$ and $w <_{\mathbb{E}^*} x \cdot y$.*

Proof. If w is not a prefix of x, then $w \cdot y \neq x$. By Lemma 44, we have that x is not a prefix of w, and thus that $w \neq x \cdot y$. It remains to be shown that $w \cdot y \leq_{\mathbb{E}^*} x$ and $w \leq_{\mathbb{E}^*} x \cdot y$.

Let u be the largest common prefix of w and x. Write $w = u \cdot w'$ and $x = u \cdot x'$; we now know that $w' \leq_{\mathbb{E}^*} x'$. Because w is not a prefix of x, w' cannot be empty. Likewise, x' cannot be empty because x is not a prefix of w.

We now show that $w \cdot y \leq_{\mathbb{E}^*} x$. Write $w' = e \cdot w''$ and $x' = f \cdot x''$. Since u is the largest common prefix of w and x, it follows that $e \neq f$ and thus we know that $e <_{\mathbb{E}} f$. But then $e \cdot w'' \cdot y \leq_{\mathbb{E}^*} f \cdot x''$ and thus $w \cdot y = u \cdot w' \cdot y = u \cdot e \cdot w'' \cdot y \leq_{\mathbb{E}^*} u \cdot f \cdot x'' = u \cdot x' = x$. The proof that $w \leq_{\mathbb{E}^*} x \cdot y$ is similar. □

Lemma 46. *Let $w, x, y, z \in \mathbb{E}^*$ such that $w <_{\mathbb{E}^*} x$ and w is not a prefix of x. Then $w \cdot y <_{\mathbb{E}^*} x \cdot z$.*

Proof. If $w <_{\mathbb{E}^*} x$ and w is not a prefix of x, then $w \cdot y <_{\mathbb{E}^*} x$ by Lemma 45. But then $w \cdot y$ is also not a prefix of x (otherwise w would be too); again by Lemma 45 we conclude that $w \cdot y <_{\mathbb{E}^*} x \cdot z$. □

Lemma 47. *Let $w, x, y \in \mathbb{E}^*$ such that $w \leq_{\mathbb{E}^*} x$ and $|w| = |x|$, then $w \cdot y \leq_{\mathbb{E}^*} x \cdot y$.*

Proof. If $w = x$, then $w \cdot y \leq_{\mathbb{E}^*} x \cdot y$ by reflexivity of $\leq_{\mathbb{E}^*}$. Otherwise we know that $w <_{\mathbb{E}^*} x$. Since $|w| = |x|$ but $w \neq x$, we know that w is not a prefix of x; then $w \cdot y \leq_{\mathbb{E}^*} x \cdot y$ follows by Lemma 46. □

Lemma 48. *Let $w, x, y, z \in \mathbb{E}^*$ such that $|w| = |x|$. If $w \cdot y \leq_{\mathbb{E}^*} x \cdot z$, then either $w = x$ and $y \leq_{\mathbb{E}^*} z$, or $w <_{\mathbb{E}^*} x$.*

Proof. We prove the claim by induction on $|w|$. If $|w| = 0$, then $w = \epsilon = x$ and $y \leq_{\mathbb{E}^*} z$ immediately. If $|w| > 0$, let $w = e \cdot w'$ and $x = f \cdot x'$ and assume that $w' \cdot y \leq_{\mathbb{E}^*} x' \cdot z$ implies $w' = x'$ and $y \leq_{\mathbb{E}^*} z$ or $w' <_{\mathbb{E}^*} x'$. If $e = f$, then $w' \cdot y \leq_{\mathbb{E}^*} x' \cdot z$ by definition of $\leq_{\mathbb{E}^*}$, and thus (making use of the induction hypothesis) either $w = e \cdot w' = f \cdot x' = x$ and $y \leq_{\mathbb{E}^*} z$, or $w = e \cdot w' <_{\mathbb{E}^*} f \cdot x' = x$. If on the other hand $e <_{\mathbb{E}} f$, then $w = e \cdot w' \leq_{\mathbb{E}^*} f \cdot x' = x$ by definition of $\leq_{\mathbb{E}^*}$, and $w \neq x$, thus $w <_{\mathbb{E}^*} x$. □

Lemma 49. *Let $w, x, y, z \in \mathbb{E}^*$. If $x \leq_{\mathbb{E}^*} y$ and $|x| = |y|$, then $w \cdot x \cdot z \leq_{\mathbb{E}^*} w \cdot y \cdot z$.*

Proof. If $x = y$, then $w \cdot x \cdot z = w \cdot y \cdot z$ and thus the claim holds immediately. If $x <_{\mathbb{E}^*} y$, then x is not a prefix of y (since $x \neq y$ and $|x| = |y|$), thus $x \cdot z <_{\mathbb{E}^*} y \cdot z$ by Lemma 46. By definition of $\leq_{\mathbb{E}^*}$ we can conclude that $w \cdot x \cdot z \leq_{\mathbb{E}^*} w \cdot y \cdot z$. □

C.2 Lemma's for the generalized lexicographic order

Lemma 50. *Let $w, x \in \mathfrak{P}(A)^*$ such that $|w| + |x| \leq p$. Then*

$$\mathsf{idle}_A(w \cdot x, p) = \bigcup_{|w| \leq k \leq p - |x|} \mathsf{idle}_A(w, k) \cdot \mathsf{idle}_A(x, p - k)$$

Proof. We start with the inclusion from left to right. Let $t \in \mathsf{idle}_A(w \cdot x, p)$. Since $\mathfrak{p}_A(t) = w \cdot x$, we can write $t = u \cdot v$ such that $\mathfrak{p}_A(u) = w$ and $\mathfrak{p}_A(v) = x$. Therefore $u \in \mathsf{idle}_A(w, |u|)$ and $v \in \mathsf{idle}_A(x, p - |u|)$ and thus $t \in \mathsf{idle}_A(w, |u|) \cdot \mathsf{idle}_A(x, p - |u|)$. Since $|w| = |\mathfrak{p}_A(u)| \leq |u| = p - |v| \leq p - |x|$, we know that $t \in \bigcup_{|w| \leq k \leq p - |x|} \mathsf{idle}_A(w, k) \cdot \mathsf{idle}_A(x, p - k)$.

Now, for the inclusion from right to left, let $t \in \mathsf{idle}_A(w, k) \cdot \mathsf{idle}_A(x, p - k)$ for some $|w| \leq k \leq p - |x|$. Then $t = u \cdot v$ with $u \in \mathsf{idle}_A(w, k)$ and $v \in \mathsf{idle}_A(x, p - k)$. Thus $\mathfrak{p}_A(t) = \mathfrak{p}_A(u) \cdot \mathfrak{p}_A(v) = w \cdot x$ and $|t| = |u| + |v| = k + p - k = p$; we conclude that $t \in \mathsf{idle}_A(w \cdot x, p)$. □

Lemma 51. *Let $w, x, y \in \mathfrak{P}(A)^*$ such that w is effective. Then $w \cdot x \sqsubseteq_A w \cdot y$ if and only if $x \sqsubseteq y$.*

Proof. We start by proving the claim from left to right. If $|x| \leq |y|$, we know that for all $u \in \mathsf{idle}_A(w \cdot x, |w \cdot y|)$ we have $u \leq w \cdot y$. Let $x' \in \mathsf{idle}_A(x, |y|)$. Then we know that $w \cdot x' \in \mathsf{idle}_A(w \cdot x, |x| + |y|)$ by Lemma 50. Therefore $w \cdot x' \leq w \cdot y$, and thus $x' \leq y$ by Lemma 48. We conclude that $x \sqsubseteq_A y$.

We argue the case where $|x| > |y|$ by induction on w. If $w = \epsilon$, then the claim holds trivially. Assume now that $w = e \cdot w'$ and that the claim holds for w'. By our premise, we know that $w \cdot x = e \cdot w' \cdot x \leq u$ for some $u \in \mathsf{idle}_A(w \cdot y, |w| + |y|)$. Write $u = f \cdot u'$. Observe that, since $\mathsf{p}_A(u) = \mathsf{p}_A(f \cdot u') = x \cdot z$, we know that f must be either e or i. Because we know that $e > t$, it follows that $f = e$ (otherwise $u = i \cdot u' < e \cdot w' \cdot x = w \cdot x$). By Lemma 48, we know that $w' \cdot x \leq u'$. Since $u' \in \mathsf{idle}_A(w' \cdot y, |w'| + |x|)$, we have that $w' \cdot x \sqsubseteq_A w' \cdot y$ from which $x \sqsubseteq y$ follows by the induction hypothesis.

We now prove the claim from right to left. Let $p = \max(|x|, |y|)$ and $t \in \mathsf{idle}_A(w \cdot x, |w| + p)$. If w is a prefix of t, then write $t = w \cdot x'$. We know that $x' \in \mathsf{idle}_A(x, p)$, and thus that there exists a $y' \in \mathsf{idle}_A(y, p)$ such that $x' \leq y'$. But then $t = w \cdot x' \leq w \cdot y' \in \mathsf{idle}_A(w \cdot y, |w| + p)$. If w is not a prefix of t, then let z be the longest common prefix of w and t. Write $w = z \cdot w'$. By construction, $w' \neq \epsilon$ and $z \cdot t$ is a prefix of t; write $t = z \cdot t \cdot u$ to reflect this. Since w is effective, so is w', from which we know that $i < w'$, thus $z \cdot t < z \cdot w' = w$ by definition of $\leq$. Choose any $y' \in \mathsf{idle}_A(y, p)$. By Lemma 46, it follows that $t = z \cdot t \cdot u \leq w \cdot y' \in \mathsf{idle}_A(w \cdot y, |w| + p)$. We can conclude that $w \cdot x \sqsubseteq_A w \cdot y$. □

Lemma 52. *Let $w \in \mathfrak{P}(A)^*$ be effective. Then $\epsilon \sqsubseteq_A w$.*

Proof. If $w = \epsilon$ we are done immediately by reflexivity of $\sqsubseteq_A$. Assume now that $w = e \cdot w'$. Let $u \in \mathsf{idle}_A(\epsilon, |w|)$, then $u = i^{|w|}$. Since $i <_E e$, it follows immediately that $u = i \cdot t^{|w'|} \leq e \cdot w' = w$. Thus $\epsilon \sqsubseteq w$. □

Lemma 53. *Let $w \in \mathfrak{P}(A)^*$ be effective. If x is a prefix of w, then $x \sqsubseteq_A w$.*

Proof. Let $w = x \cdot w'$. By Lemma 52, we know that $\epsilon \sqsubseteq_A w'$, and by Lemma 51, we conclude $x \sqsubseteq x \cdot w' = w$. □

Lemma 54. *Let $w, x, y \in \mathfrak{P}(A)^*$ such that $w \sqsubseteq_A x$. If $|w| \geq |x|$ and $w \neq x$, then $w \cdot y \sqsubseteq x$.*

Proof. We know that there exists some $x' \in \mathsf{idle}_A(x, |w|)$ such that $w < x'$ (equality is ruled out by the premises). Then $w \cdot y < x' \cdot t^{|y|}$ by Lemma 46. Since $x' \cdot t^{|y|} \in \mathsf{idle}_A(x, |w \cdot y|)$, we then know that $w \cdot y \sqsubseteq_A x$. □

Lemma 55. *Let $w, x, y \in \mathfrak{P}(A)^*$ such that $w \sqsubseteq_A x$. If $|w| > |x|$, then $w \cdot y \sqsubseteq x \cdot y$.*

Proof. We prove the claim by induction on y. If $y = \epsilon$ then the claim holds immediately. Assume now that $y = e \cdot y'$ and that the claim holds for y'. We know that there exists an $x' \in \mathsf{idle}_A(x, |w|)$ such that $w \leq x'$. Choose $x'' = x' \cdot e$. Then $x'' \in \mathsf{idle}_A(x \cdot e, |w| + 1)$, and by Lemma 47, we know that $w \cdot e \leq x' \cdot e$. Therefore $w \cdot e \sqsubseteq_A x \cdot e$. Since $|x| + 1 < |w| + 1$, we immediately have that $w \cdot y = w \cdot e \cdot y' \sqsubseteq x \cdot e \cdot y' = x \cdot y$ by our induction hypothesis. □

Lemma 56. *Let $w, x, y \in \mathfrak{P}(A)^*$ such that $w \sqsubseteq_A x$, $|w| \leq |x|$, but w is not a prefix of x. Then $w \cdot y \sqsubseteq x \cdot y$.*

Proof. Let $t \in \mathsf{idle}_A(w \cdot y, |x \cdot y|)$. We need to show that $t \leq x \cdot y$. By Lemma 50, we know that $t = u \cdot v$ for $u \in \mathsf{idle}_A(w, k)$ and $v \in \mathsf{idle}_A(y, |x \cdot y| - k)$, with $|w| \leq k \leq |x|$. Then choose $w' = u \cdot t^{|x| - k}$. Since $w' \in \mathsf{idle}_A(w, |x|)$ and $w \sqsubseteq_A x$, we know that $w' \leq x$. Since w is not a prefix of x, neither is u. Write $x = r \cdot s$ such that $|r| = |u|$. Since $u \cdot t^{|x|-k} \leq r \cdot s$ and $u \neq r$, it follows by Lemma 48 that $u < r$. Consequently, by Lemma 46, we have $t = u \cdot v \leq r \cdot s \cdot y = x \cdot y$ and thus $w \cdot y \sqsubseteq x \cdot y$. □

Lemma 57. *Let $w, x, y \in \mathfrak{P}(A)^*$ such that $w \sqsubseteq_A x$. If w is not a prefix of x, then $w \cdot y \sqsubseteq x \cdot y$.*

Proof. Either $|w| > |x|$ or $|w| \leq |x|$. In the former case the claim follows from Lemma 55, while in the latter case the claim follows from Lemma 56 □

Lemma 58. *Let $w, x \in \mathfrak{P}(A)^*$ such that w and x are effective. Then $w \leq x$ if and only if $w \sqsubseteq_A x$.*

Proof. We isolate a few special cases first. If w is a prefix of x, then $w \leq x$ by Lemma 43 and $w \sqsubseteq_A x$ by Lemma 53, and so the claim holds immediately.

If on the other hand x is a strict prefix of w, then write $w = x \cdot z$. Assume towards a contradiction that $w \leq x$. Then $x \cdot z \leq x$ implies that $z \leq \epsilon$ by Lemma 48. But then $z = \epsilon$ by antisymmetry, which is a contradiction because x is a strict prefix of w. We thus know that $w \not\leq x$. Similarly, assume that $x \cdot z = w \sqsubseteq_A x$. Then, by Lemma 51, we know that $z \sqsubseteq \epsilon$ and (since $\epsilon \sqsubseteq z$ by Lemma 52) that $z = \epsilon$, again contradicting that x is a strict prefix of w. We therefore know that $w \not\sqsubseteq x$. Since both $w \not\leq x$ and $w \not\sqsubseteq x$ when x is a strict prefix of w, the claim holds in this case too.

Assume for the remainder that w is not a prefix of x and x is not a prefix of w. Let z be the longest common prefix of w and x. Write $w = z \cdot w'$ and $x = z \cdot x'$. Observe that neither w' nor x' is empty, for otherwise w

(respectively x) would be a prefix of x (respectively w). Write e for the first position of w' and f for the first position of x' and note that $e \neq f$. Choose $p = \max(|w'|, |x'|)$.

For the direction from left to right, assume that $w \leq x$. Then $w' \leq x'$ by Lemma 48; since $e \neq f$ we can also derive that $e < f$ by Lemma 48. Let $w'' \in \mathsf{idle}_A(w', p)$. If w'' starts with i, then $p = |x'|$ and we know that $w'' < x' \in \mathsf{idle}_A(x', p)$. If w'' does not start with i, it starts with e. But then $w'' < x' \cdot t^{p-|x|} \in \mathsf{idle}_A(x', p)$. At any rate, $w' \sqsubseteq_A x'$. But then, by Lemma 51, we can derive that $w = z \cdot w' \sqsubseteq z \cdot x' = x$.

For the direction from right to left, let $w \sqsubseteq_A x$. Then $z \cdot w' \sqsubseteq z \cdot x'$, so by Lemma 51 we see that $w' \sqsubseteq x'$. If $p = |x'|$, then $w' \cdot t^{|w'|-|x'|} \leq x'$. If $p = |w'|$, then $w' \leq x''$ for some $x'' \in \mathsf{idle}_A(x', p)$. Note that x'' cannot start with i, because then $w' > x''$ since w' is effective; thus x'' starts with f. In either case, we can derive from $e \neq f$ and Lemma 48 that $e < f$, and thus $w' < x'$, from which we have that $w = z \cdot w' \leq z \cdot x' = x$. □

Lemma 59. *Let $x, y, z \in \mathfrak{P}(A)^*$ such that $x \leq y \cdot x$ and y is effective. Then $x \sqsubseteq_A y \cdot x$*

Proof. First, observe that the claim holds for $y = \epsilon$. We prove the claim for $y = f \cdot y'$ by induction on x.

For the base case, where $x = \epsilon$, the claim holds by virtue of Lemma 52. Now write $x = e \cdot x'$ and assume the claim holds for x'. If $e \cdot x' \leq f \cdot y' \cdot e \cdot x'$, then either $e < f$, or $e = f$ and $x' \leq y' \cdot e \cdot x'$ by Lemma 48.

If $e < f$, then let $z \in \mathsf{idle}_A(x, |y \cdot x|)$ and write $z = g \cdot z'$. Whether $g = i$ or $g = e$, we can derive that $z = g \cdot z < f \cdot y' \cdot x$, allowing us to derive that $x \sqsubseteq_A y \cdot x$ in this case. If $e = f$, then $y' \cdot e$ is effective, and so by the induction hypothesis we know that $x' \sqsubseteq y' \cdot e \cdot x'$. But then by Lemma 51, $x = e \cdot x' \sqsubseteq f \cdot y' \cdot e \cdot x' = y \cdot x$. □

Lemma 60. *Let $w, x, y \in \mathfrak{P}(A)^*$ such that w and x are effective and $w \cdot y \leq x \cdot y$. Then $w \cdot y \sqsubseteq_A x \cdot y$.*

Proof. If $|w| = |x|$ then the claim holds immediately. If $|w| > |x|$, then $w \cdot y$ is not a prefix of $x \cdot y$. Then by Lemma 45 we know that $w \cdot y \leq x \cdot y \cdot t^{|w|-|x|}$. The latter sequence is contained in $\mathsf{idle}_A(x \cdot y, |w \cdot y|)$, and thus we have that $w \cdot y \sqsubseteq_A x \cdot y$.

If $|w| < |x|$, write $x = t \cdot u$ such that $|t| = |w|$. Now, since $w \cdot y \leq t \cdot u \cdot y$, by Lemma 48 we have either $w = t$ and $y \leq u \cdot y$, or $w < t$. In the former case, u is effective and therefore by Lemma 59 it follows that $y \sqsubseteq_A u \cdot y$, from which we have $w \cdot y \sqsubseteq t \cdot u \cdot y = x \cdot y$ by Lemma 51, since $w = t$ is effective.

In the case where $w < t$, we know that w is not a prefix of t and thus that $w < t \cdot u = x$ by Lemma 45. Note that since $|w| = |t|$, w is also not a prefix of x. Since w and x are effective, we know by Lemma 58 that $w \sqsubseteq_A x$. Then, by Lemma 57 we have that $w \cdot y \sqsubseteq x \cdot y$. □

Lemma 61. *Let $w \in \mathfrak{P}(A)^*$; w is non-effective if and only if $w = e \cdot w'$ such that $i \not\leq e$ or w' is non-effective.*

Proof. If w is non-effective, then $w \neq \epsilon$. Write $w = e \cdot w'$. If $i < e$ and w' is effective, then w is effective, too. Thus either $i \not\leq e$ or w' is non-effective. □

Lemma 62. *Let w be non-effective and let x be a non-effective proper prefix of w. Then $x \not\sqsubseteq_A w$*

Proof. Write $w = x \cdot t$. Let z be the longest effective prefix of x; write $x = z \cdot u$. Observe that u must be non-empty (otherwise $x = z$ would be effective) and that if the first position of u is e, then $i \not\leq e$ (otherwise z could be longer). Now choose $x' = z \cdot t^{|w|-|x|} \cdot u \in \mathsf{idle}_A(x, |w|)$. Because $|w| > |x|$, we know that $i^{|w|-|x|} \cdot u \not\leq u \cdot t$ and thus we can derive $x' = z \cdot t^{|w|-|x|} \cdot u \not\leq z \cdot u \cdot t = w$. It follows that $x \not\sqsubseteq_A w$. □

Lemma 63. *Let $w, x, y \in \mathfrak{P}(A)^*$ such that w is non-effective and $|x| < |y|$. Then $w \cdot x \not\sqsubseteq_A w \cdot y$.*

Proof. Note that we can disregard the case for $w = \epsilon$, for ϵ is effective. For the remainder, it suffices to prove that $z \not\leq w \cdot y$ for some $z \in \mathsf{idle}_A(w \cdot x, |w| + |y|)$.

Let $w = z \cdot w'$ such that z is the longest effective prefix of w. Since w is non-effective, w' is non-effective. Then $w' = e \cdot w''$ with $i \not\leq e$ (otherwise z could be longer). Assume towards a contradiction that $w' \cdot x \not\sqsubseteq_A w' \cdot y$. Then in particular $i^{|y'|-|x'|} \cdot w' \cdot x \leq w' \cdot y$. By Lemma 48 and the fact that $|y'| > |x'|$, this would imply that $i \leq e$, which is a contradiction. We thus conclude that $w' \cdot x \not\sqsubseteq w' \cdot y$. Consequently, we know that $w \cdot x = z \cdot w' \cdot x \not\sqsubseteq z \cdot w' \cdot y = w \cdot y$ by Lemma 51. □

Lemma 64. *Let $w, x, y \in \mathfrak{P}(A)^*$ such that $w \cdot x \leq x \cdot w$, $y \sqsubseteq_A x$ and $|y| \leq |x|$. Furthermore, let w be non-effective and let x be an effective prefix of of w. Then $w \cdot y \sqsubseteq x \cdot w$.*

Proof. As a special case, note that when x is empty, y must also be empty and then the claim holds immediately. We assume that $x \neq \epsilon$ for the remainder of this proof. Let $z \in \mathsf{idle}_A(w \cdot y, |x \cdot w|)$. Since $|w \cdot y| \leq |x \cdot w|$, our objective is to show that $z \leq x \cdot w$.

If w is a prefix of z, then by Lemma 50 $z = w \cdot y'$ for some $y' \in \mathsf{idle}_A(y, |x|)$. Since $y \sqsubseteq_A x$ we know that $y' \leq x$ and therefore $z = w \cdot y' \leq w \cdot x \leq x \cdot w$.

If w is not a prefix of z, then we can write $w = t \cdot t \cdot u$ such that t is a proper prefix of w; consequently, write $w = t \cdot v$. We also factor $w = p \cdot e \cdot q$ such that p is the longest effective prefix of w — this is possible because w

is non-effective. Note that e is non-effective and, since $|x| \geq 1$ and x is effective, we know that the first $|p|+1$ positions of $x \cdot w$ are effective.

Now that t and p are both prefixes of w, one must be a prefix of the other.

In cases where t is a proper prefix of p, we know that $t < p$ by Lemma 43. Since p is effective, we can derive that $t \cdot t < p$. Now $t \cdot t$ cannot be a prefix of p, and so we can derive by Lemma 46 that $z = t \cdot t \cdot u < p \cdot e \cdot q \cdot x = w \cdot x$.

When p is a prefix of t, write $x \cdot w = r \cdot s$ such that $|r| = |p| + 1$; note that r is effective and therefore not a prefix of $p \cdot e$. Assume towards a contradiction that $r < p \cdot e$. Then by Lemma 46, $x \cdot w = r \cdot s < p \cdot e \cdot q \cdot x = w \cdot x$, contradicting our assumption — thus $p \cdot e < r$. If $t = p$, then $p \cdot t < r$ since r is effective; it follows by Lemma 46 that $z = t \cdot t \cdot u = p \cdot t \cdot u < r \cdot s = x \cdot w$. If p is a proper prefix of t, then $p \cdot e$ is a prefix of t (since both are prefixes of w), and so $t < r$ by Lemma 45, from which we derive (again by Lemma 46) that $z = t \cdot t \cdot u < r \cdot s = x \cdot w$. □

Lemma 65. *Let $w, x \in \mathfrak{P}(A)^*$ such that w is non-empty and effective. If $w \cdot x = x \cdot w$, then x is effective.*

Proof. If $|w| \geq |x|$, then $w \cdot x = x \cdot w$ implies that the first $|w|$ positions of $x \cdot w$ are effective; this includes the positions of x and therefore x is effective.

We prove the case for $|w| < |x|$ by induction on x. For the base case, we have $x = e \cdot f$. Since $|w| < |x|$ and w is non-empty, it follows that $w = g$ for $g > t$. Since $g \cdot e \cdot f = w \cdot x = x \cdot w = e \cdot f \cdot g$, we know that $g = f$ and $g = e$, making $e \cdot f = x$ effective. For the inductive step, write $x = y \cdot z$ such that $|y| = |z|$ and assume the claim holds for z. Then $w \cdot y \cdot z = w \cdot x = x \cdot w = y \cdot z \cdot w$. From this, we learn that $y \cdot z = z \cdot w$ and $w = y$; therefore $w \cdot z = z \cdot w$. If $|w| \geq |z|$ then we know that z is effective by the reasoning above, otherwise it follows from the induction hypothesis. Since $w = y$, y is also effective, making $x = y \cdot z$ effective as well. □

C.3 Proofs of Theorem 3 and Theorem 4

Theorem 3. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA. Let $L \subseteq \mathfrak{P}(A)^*$ be non-empty and finite, and let $\leq_{\mathbb{E}^*}$ be total. Let $w = \max_{\sqsubseteq_A}(L)$ and choose the longest $x \in \mathsf{prefix}(w) \cap L$ such that $x \cdot w$ is the $\leq_{\mathbb{E}^*}$-maximum of $(\mathsf{prefix}(w) \cap L) \cdot w$.[15] If w is effective or x is a proper prefix of w, then $\max_{\sqsubseteq_A}(L^n) = x^{n-1} \cdot w$ for $n \geq 1$.*

Proof. Since x is a prefix of w, we can write $w = x \cdot y$. It will now suffice to prove that $\max_{\sqsubseteq_A}(L^n) = x^n \cdot y$. We prove the claim by induction on n. For the base case, where $n = 1$, observe that $\max_{\sqsubseteq_A}(L^1) = w = x^1 \cdot y$.

Assume that the claim holds for n; we need to prove it for $n+1$. Write $\max_{\sqsubseteq_A}(L^{n+1}) = t \cdot u$ such that $t \in L$ and $u \in L^n$. The inductive step is divided into three parts. First, we show that t must be an effective prefix of w. Then, we prove that $u = x^n \cdot y$. Finally, we argue that $t = x$.

Now assume towards a contradiction that t is not a prefix of w. First, note that $t \sqsubseteq_A w$ since $t \in L$. Then from $t \sqsubseteq_A w$ and Lemma 57 we have that $t \cdot u \sqsubseteq_A w \cdot u$. But then, since $w \cdot u \sqsubseteq_A \max(L^{n+1}) = t \cdot u$, we have that $t \cdot u = w \cdot u$, which is a contradiction because it implies that $t = w$, making t a prefix of w.

If w is effective, then t is effective, too. If w is non-effective, we need to rule out only the possibility that $t = w$; if t is a proper prefix of w, then t is effective by Lemma 62. Note that, in this case, also by Lemma 62, we know that x must be effective (otherwise $x \not\sqsubseteq_A w$). Assume for the remainder of this paragraph that $t = w$. Write $u = r \cdot s$ such that $r \in L$ and $s \in L^{n-1}$ — this is possible since $n \geq 1$. Because $w \cdot x \cdot s \sqsubseteq_A w \cdot r \cdot s$, by Lemma 63 it follows that $|r| \leq_{\mathbb{E}^*} |x|$. From this, we can also derive that $r \sqsubseteq_A x$: if r is a prefix of x then r is effective and so $r \sqsubseteq_A x$ immediately by Lemma 53; if r is not a prefix of x, then $r \sqsubseteq_A x$ (for otherwise $x \sqsubset r$ would imply that $w \sqsubset r$ by Lemma 54, contradicting $r \sqsubseteq_A w$). Lastly, note that since $w \cdot w \leq_{\mathbb{E}^*} x \cdot w$, it follows by Lemma 48 that $w \cdot x \leq_{\mathbb{E}^*} x \cdot w$. We now have all premises in place to invoke Lemma 64 and conclude that $w \cdot r \sqsubseteq_A x \cdot w$, thus $w \cdot r \cdot s \sqsubseteq_A x \cdot w \cdot s$ by Lemma 57. Since $x \cdot w \cdot s \sqsubseteq_A w \cdot r \cdot s$, we know that $w \cdot r \cdot s = x \cdot w \cdot s$; it follows that we can choose $t \in L$ and $u \in L^n$ such that $\max_{\sqsubseteq_A}(L^{n+1}) = t \cdot u$ and t is effective. For the remainder of this proof we can therefore safely assume that t is effective.

To show that $u = x^n \cdot y$, first observe that, since $t \cdot u = \max_{\sqsubseteq_A}(L^{n+1})$ and $t \cdot \max_{\sqsubseteq_A}(L^n) \in L^{n+1}$, we know that $t \cdot \max_{\sqsubseteq_A}(L^n) \sqsubseteq_A t \cdot u$. But then, because t is effective, by Lemma 51 we know that $\max_{\sqsubseteq_A}(L^n) \sqsubseteq_A u$ and therefore $u = \max_{\sqsubseteq_A}(L^n)$ since $u \in L^n$. By the induction hypothesis, we then know that $u = x^n \cdot y$.

It remains to be shown that $t = x$. Since both t and x are prefixes of w, either t is a prefix of x, or vice versa.

If t is a prefix of x, then write $x = t \cdot x'$. Then, since $w \cdot x \leq_{\mathbb{E}^*} x \cdot w$, we can derive that $t \cdot t \cdot x' \cdot y = t \cdot x \cdot y \leq_{\mathbb{E}^*} x \cdot x \cdot y = t \cdot x' \cdot t \cdot x' \cdot y$, and thus that $t \cdot x' \cdot y \leq_{\mathbb{E}^*} x' \cdot t \cdot x' \cdot y$ by Lemma 48. But then, again by Lemma 48, we know that $t \cdot x' \leq_{\mathbb{E}^*} x' \cdot t$. Consequently, we know that $(t \cdot x')^n \cdot y \leq_{\mathbb{E}^*} (x' \cdot t)^n \cdot x' \cdot y$, and thus that $t \cdot (t \cdot x')^n \cdot y \leq_{\mathbb{E}^*} t \cdot (x' \cdot t)^n \cdot x' \cdot y$. From this, we have that $t \cdot x^n \cdot y \leq_{\mathbb{E}^*} (t \cdot x')^{n+1} \cdot y = x^{n+1} \cdot y$.

If x is a prefix of t, write $w = t \cdot w'$. Then $t \cdot w \leq_{\mathbb{E}^*} x \cdot w$, and so $t \cdot x \cdot y \leq_{\mathbb{E}^*} x \cdot t \cdot w'$. But then $t \cdot x \leq_{\mathbb{E}^*} x \cdot t$ by Lemma 48. From this, we can derive that $t \cdot x^n \cdot y \leq_{\mathbb{E}^*} x^{n-1} \cdot t \cdot x \cdot y$ by repeated application of Lemma 49. Since $t \cdot w \leq_{\mathbb{E}^*} x \cdot x \cdot y$, we also have that $x^{n-1} \cdot t \cdot x \cdot y \leq_{\mathbb{E}^*} x^{n-1} \cdot x \cdot x \cdot y = x^{n+1} \cdot y$, and so $t \cdot x^n \cdot y \leq_{\mathbb{E}^*} x^{n+1} \cdot y$.

In either case $t \cdot x^n \cdot y \leq_{\mathbb{E}^*} x^{n+1} \cdot y$. Because t and x are effective, we can derive by Lemma 60 that $t \cdot x^n \cdot y \sqsubseteq_A x^{n+1} \cdot y$. Since $x^{n+1} \cdot y \sqsubseteq_A t \cdot u = t \cdot x^n \cdot y$, it follows that $t \cdot x^n \cdot y = x^{n+1} \cdot y$, and therefore $t = x$. In conclusion, we have that $\max(L^{n+1}) = t \cdot u = x \cdot x^n \cdot y = x^{n+1} \cdot y = x^n \cdot w$, thus establishing the theorem. □

[15] Such an x always exists, for $w \cdot w \in (\mathsf{prefix}(w) \cap L) \cdot w$ and $(\mathsf{prefix}(w) \cap L) \cdot w$ is finite.

Theorem 4. *Let $A = \langle Q, \Sigma, \mathbb{E}, \rightarrow, q^0, t \rangle$ be a threshold-free SCA. Let $L \subseteq \mathfrak{P}(A)^*$ be non-empty and finite, and let $\leq_{\mathbb{E}^*}$ be total. Let $w = \max_{\sqsubseteq_A}(L)$. If w is non-effective and $w \cdot w$ is the $\leq_{\mathbb{E}^*}$-maximum of $(\mathsf{prefix}(w) \cap L) \cdot w$, then $\max_{\sqsubseteq_A}(L^n) = w \cdot z^{n-1}$ for $n \geq 1$, where z is the $\leq_{\mathbb{E}^*}$-maximum of the shortest elements in L.*

Proof. We prove the claim by induction on n. For our base case, where $n = 1$, the claim holds immediately. For the inductive step, assume that the claim holds for n. Write $\max_{\sqsubseteq_A}(L^{n+1}) = t \cdot u$ such that $t \in L^n$ and $u \in L$; we now need to show that $t \cdot u = w \cdot z^n$.

First, assume towards a contradiction that $t \cdot u = \epsilon$. If this is the case, then $\epsilon \in L$. However, $\epsilon \sqsubseteq_A w$ since $\max_{\sqsubseteq_A}(L) = w$. Since w is non-effective, we moreover know that $w \neq \epsilon$. But then $t \cdot u \sqsubseteq_A w \cdot \epsilon^n \in L^{n+1}$, contradicting that $\max_{\sqsubseteq_A}(L^{n+1}) = t \cdot u$. We thus know that $t \cdot u \neq \epsilon$. We can therefore assume without loss of generality that $t \neq \epsilon$, choosing $u = \epsilon$ if necessary.

Assume towards a contradiction that $t \neq w \cdot z^{n-1}$. Then $t \sqsubset w \cdot z^{n-1}$ by the induction hypothesis. If t is not a prefix of $w \cdot z^{n-1}$, then $t \cdot u \sqsubset w \cdot z^{n-1} \cdot u$ by Lemma 57, contradicting that $t \cdot u$ is the $\sqsubseteq_A$-maximum of L^{n+1}. If t is a proper prefix of $w \cdot z^{n-1}$, then t is either effective or non-effective. If, on the one hand, t is non-effective, then $t \not\sqsubseteq_A w \cdot z^{n-1}$ by Lemma 62, which is a contradiction. If, on the other hand, t is effective, then t is a proper prefix of w. But then $t \cdot w <_{\mathbb{E}^*} w \cdot w$ and so $t \cdot w \leq_{\mathbb{E}^*} w \cdot t$. If $t \cdot w = w \cdot t$, then w is effective by Lemma 65, since t is non-empty and effective. Therefore $t \cdot w <_{\mathbb{E}^*} w \cdot t$ and thus $t \cdot w \sqsubset w \cdot t$. Because $u \sqsubseteq_A w$, we moreover know that $t \cdot u \sqsubseteq_A t \cdot w$ by Lemma 51. By transitivity, we then know that $t \cdot u \sqsubset w \cdot t \in L^{n+1}$. With the latter observation, we have again reached a contradiction. We thus surmise that $t = w \cdot z^{n-1}$.

Observe that t is non-effective. If $|u| > |z|$, then $t \cdot u \sqsubset t \cdot z$ by Lemma 63, again contradicting that $\max_{\sqsubseteq_A}(L^{n+1}) = t \cdot u$. We therefore know that $|u| \leq_{\mathbb{E}^*} |z|$. Since no element of L is shorter than z, it follows that $|u| = |z|$. If $u <_{\mathbb{E}^*} z$, then $t \cdot u <_{\mathbb{E}^*} t \cdot z$ and again $t \cdot u \sqsubset t \cdot z$; thus $u \geq z$. Since $u \leq_{\mathbb{E}^*} z$, it follows that $u = z$.

In conclusion, we know that $t = w \cdot z^{n-1}$ and $u = z$, and therefore $\max_{\sqsubseteq_A}(L^{n+1}) = t \cdot u = w \cdot z^{n-1} = w \cdot z^n$, thus establishing the claim. □

References

[1] Farhad Arbab and Francesco Santini. Preference and Similarity-Based Behavioral Discovery of Services. In *Proc. Web Services and Formal Methods (WS-FM)*, pages 118–133, 2012. URL: http://dx.doi.org/10.1007/978-3-642-38230-7_8.

[2] Christel Baier, Tobias Blechmann, Joachim Klein, and Sascha Klüppelholz. Formal verification for components and connectors. In *Proc. Formal Methods for Components and Objects*, pages 82–101, 2008. URL: http://dx.doi.org/10.1007/978-3-642-04167-9_5.

[3] Christel Baier, Tobias Blechmann, Joachim Klein, Sascha Klüppelholz, and Wolfgang Leister. Design and verification of systems with exogenous coordination using Vereofy. In *Proc. International Symposium on Leveraging Applications, ISoLA 2010*, pages 97–111, 2010. URL: http://dx.doi.org/10.1007/978-3-642-16561-0_15.

[4] Christel Baier, Marjan Sirjani, Farhad Arbab, and Jan Rutten. Modeling component connectors in Reo by constraint automata. *Science of Computer Programming*, 61:75–113, 2006. URL: http://dx.doi.org/10.1016/j.scico.2005.10.008.

[5] Vijay G. Bharadwaj and John S. Baras. Towards automated negotiation of access control policies. In *Proc. International Workshop on Policies for Distributed Systems (POLICY)*, pages 111–119, 2003. URL: http://dx.doi.org/10.1109/POLICY.2003.1206965.

[6] Stefano Bistarelli. *Semirings for Soft Constraint Solving and Programming*, volume 2962 of *Lecture Notes in Computer Science*. Springer, 2004. URL: http://dx.doi.org/10.1007/b95712.

[7] Stefano Bistarelli, Hélène Fargier, Ugo Montanari, Francesca Rossi, Thomas Schiex, and Gérard Verfaillie. Semiring-based CSPs and valued CSPs: Basic properties and comparison. In *Over-Constrained Systems*, pages 111–150, 1995. URL: http://dx.doi.org/10.1007/3-540-61479-6_19.

[8] Stefano Bistarelli and Fabio Gadducci. Enhancing constraints manipulation in semiring-based formalisms. In *Proc. European Conference on Artificial Intelligence (ECAI)*, pages 63–67, 2006.

[9] Stefano Bistarelli, Ugo Montanari, and Francesca Rossi. Constraint solving over semirings. In *Proc. International Joint Conference on Artificial Intelligence, IJCAI 95*, pages 624–630, 1995.

[10] Stefano Bistarelli, Ugo Montanari, and Francesca Rossi. Semiring-based constraint satisfaction and optimization. *J. ACM*, 44(2):201–236, 1997. URL: http://dx.doi.org/10.1145/256303.256306.

[11] Julius Richard Büchi. On a decision method in restricted second order arithmetic. In *Proc. Logic, Methodology and Philosophy of Science*, pages 1–11, Stanford, Calif., 1962. Stanford Univ. Press.

[12] Rance Cleaveland and Matthew Hennessy. Priorities in process algebras. *Inf. Comput.*, 87(1/2):58–77, 1990. URL: http://dx.doi.org/10.1016/0890-5401(90)90059-Q.

[13] Rance Cleaveland, Gerald Lüttgen, and V. Natarajan. Priority and abstraction in process algebra. *Inf. Comput.*, 205(9):1426–1458, 2007. URL: http://dx.doi.org/10.1016/j.ic.2007.05.001.

[14] Fabio Gadducci, Matthias M. Hölzl, Giacoma Valentina Monreale, and Martin Wirsing. Soft constraints for lexicographic orders. In *Advances in Artificial Intelligence and Its Applications, Mexican International Conference on Artificial Intelligence, MICAI*, pages 68–79, 2013. URL: http://dx.doi.org/10.1007/978-3-642-45114-0_6.

[15] Xuechong Guan and Yongming Li. On conditions for mappings to preserve optimal solutions of semiring-induced valuation algebras. *Theor. Comput. Sci.*, 563:86–98, 2015. URL: http://dx.doi.org/10.1016/j.tcs.2014.10.016.

[16] David Harel, Dexter Kozen, and Jerzy Tiuryn. *Dynamic logic.* MIT press, 2000.

[17] Matthias M. Hölzl, Max Meier, and Martin Wirsing. Which soft constraints do you prefer? *Electr. Notes Theor. Comput. Sci.*, 238(3):189–205, 2009. URL: http://dx.doi.org/10.1016/j.entcs.2009.05.020.

[18] Tobias Kappé, Farhad Arbab, and Carolyn Talcott. A Compositional Framework For Preference-Aware Agents. In *Proc. Verification and Validation of Cyber-Physical Systems*, 2016. To appear.

[19] Tobias Kappé, Farhad Arbab, and Carolyn Talcott. A Compositional Framework For Preference-Aware Agents. CWI Technical Report FM-1603, May 2016. URL: https://repository.cwi.nl/noauth/search/fullrecord.php?publnr=24625.

[20] Christian Koehler and Dave Clarke. Decomposing port automata. In *Proc. ACM Symposium on Applied Computing (SAC)*, pages 1369–1373, 2009. URL: `http://doi.acm.org/10.1145/1529282.1529587`.

[21] Sanjiang Li and Mingsheng Ying. Soft constraint abstraction based on semiring homomorphism. *Theor. Comput. Sci.*, 403(2-3):192–201, 2008. URL: `http://dx.doi.org/10.1016/j.tcs.2008.03.029`.

[22] Alberto Lluch-Lafuente and Ugo Montanari. Quantitative μ-calculus and CTL defined over constraint semirings. *Theor. Comput. Sci.*, 346(1):135–160, 2005. URL: `http://dx.doi.org/10.1016/j.tcs.2005.08.006`.

[23] Fabio Martinelli, Ilaria Matteucci, and Francesco Santini. Semiring-based specification approaches for quantitative security. In *Proc. Quantitative Aspects of Programming Languages and Systems, QAPL*, pages 95–109, 2015. URL: `http://dx.doi.org/10.4204/EPTCS.194.7`.

[24] Amir Pnueli. The temporal logic of programs. In *Proc. Symposium on Foundations of Computer Science (SFCS)*, pages 46–57, 1977. URL: `http://dx.doi.org/10.1109/SFCS.1977.32`.

[25] Vaughan R. Pratt. Semantical considerations on Floyd-Hoare logic. In *Proc. Annual Symposium on Foundations of Computer Science*, pages 109–121, 1976. URL: `http://dx.doi.org/10.1109/SFCS.1976.27`.

[26] Jan J. M. M. Rutten. A coinductive calculus of streams. *Mathematical Structures in Computer Science*, 15(1):93–147, 2005. URL: `http://dx.doi.org/10.1017/S0960129504004517`.

[27] Thomas Schiex, Hélène Fargier, and Gérard Verfaillie. Valued constraint satisfaction problems: Hard and easy problems. In *Proc. International Joint Conference on Artificial Intelligence, IJCAI 95*, pages 631–639, 1995.

[28] Jeffrey O. Shallit. *A Second Course in Formal Languages and Automata Theory*. Cambridge University Press, 2008. URL: `http://www.cambridge.org/gb/knowledge/isbn/item1173872/?site_locale=en_GB`.

[29] Rivi Sherman, Amir Pnueli, and David Harel. Is the interesting part of process logic uninteresting? A translation from PL to PDL. *SIAM J. Comput.*, 13(4):825–839, 1984. URL: `http://dx.doi.org/10.1137/0213051`.

[30] A. Prasad Sistla, Moshe Y. Vardi, and Pierre Wolper. The complementation problem for büchi automata with applications to temporal logic (extended abstract). In *Proc. Automata, Languages and Programming*, pages 465–474, 1985. URL: `http://dx.doi.org/10.1007/BFb0015772`.

[31] Carolyn L. Talcott, Farhad Arbab, and Maneesh Yadav. Soft agents: Exploring soft constraints to model robust adaptive distributed cyber-physical agent systems. In *Software, Services, and Systems — Essays Dedicated to Martin Wirsing on the Occasion of His Retirement from the Chair of Programming and Software Engineering*, pages 273–290, 2015. URL: `http://dx.doi.org/10.1007/978-3-319-15545-6_18`.

[32] Johan van Benthem and Fenrong Liu. Dynamic logic of preference upgrade. *Journal of Applied Non-Classical Logics*, 17(2):157–182, 2007. URL: `http://dx.doi.org/10.3166/jancl.17.157-182`.

[33] Moshe Y. Vardi. An automata-theoretic approach to linear temporal logic. In *Proc. Logics for Concurrency - Structure versus Automata (8th Banff Higher Order Workshop)*, pages 238–266, 1995. URL: `http://dx.doi.org/10.1007/3-540-60915-6_6`.

[34] Soo Yeong Yi and Myung Jin Chung. Robustness of fuzzy logic control for an uncertain dynamic system. *IEEE Trans. Fuzzy Systems*, 6(2):216–225, 1998. URL: `http://dx.doi.org/10.1109/91.669018`.